HADAMARD TRANSFORM CODING OF TELEVISION SIGNALS

A theoretical investigation of the adaptive coding of Hadamard transformed television signals. The use of computable objective measures for the assessment of local subpicture characteristics in selecting appropriate coders.

By

Ibrahim Zakaria MORSI, M.Sc.

A thesis submitted to the University of Bradford
for the Degree of Doctor of Philosophy

Postgraduate School of Studies in
Electrical and Electronic Engineering
University of Bradford
Bradford, West Yorkshire,
United Kingdom.

October 1980

ACKNOWLEDGEMENT

I wish to express my thanks and deep gratitude to Dr. O.J.Downing for his advice, guidance and encouragement throughout the course of this work.

I would also like to thank Dr. P.L.M.Thorpe and Mr. J.E.Tully, former colleagues at the Digital Communication Laboratory, Bradford University. The financial support of Egyptian government is greatly acknowledged.

..... to the Greatest Two Ladies in my life,

Mother and Wife.

ABSTRACT

The problem of determining an objective means for assessing local characteristics of television subpictures in a Hadamard transform multicoder scheme is stated and discussed. Detailed investigations of transform domain coefficient statistical characteristics for different test images have been conducted. Both monochrome and colour signals were used, as well as different transform sizes and shapes.

" Directing Indexes " are proposed which, depending on the inter-relationships among transform coefficients and groups of coefficients, direct each subpicture to the appropriate coder. Three indexes in the case of monochrome signals are proposed, each with its own computational procedures and application requirements. Necessary modifications and changes for application of some indexes on colour signals are also discussed. The proposed technique of indexing eliminates the necessity of equal distribution of subpictures among 'activity classes', a major disadvantage encountered in present activity index. Coders to be used with each directing index are devised and tested, subject to an arbitrary bit rate of 2 bits per pixel, with satisfactory performance compared with some published results for other techniques.

CONTENTS

ABBREVIATIONS AND SYMBOLS

$f_{samp.}$	Sampling frequency.
f_{sc}	Chrominance subcarrier frequency.
H	Hadamard transform coefficient.
Hi or H_i	Alternative representations of the i^{th} coefficient, i being the order of coefficient.
N	Transform size (Number of samples in spatial domain, or number of coefficients in transform domain).
n	Number of bits of the binary word representing the transform size, i.e. $N=2^n$.

CHAPTER ONE

HADAMARD TRANSFORM AND BIT RATE REDUCTION

The Hadamard transform is defined, and its major characteristics are discussed with particular reference to its simple computational requirements. The transform has, as have most linear transforms, the advantage of compacting most of the image energy in fewer coefficients. Hence, bit rate reduction of digital video signal is possible by transmitting some coefficients of the transformed image rather than the image itself.

Some techniques, currently use Hadamard transform for bit rate reduction, are critically reviewed with emphasis on the adaptivity of coding schemes to the characteristics of the images coded.

1.1. Introduction

During the past decade there have been rapidly increasing studies aimed at the use of linear transformation of image signals to reduce the bandwidth required for digital transmission. This was initiated by the desire to apply digital transmission techniques to the area of television. For satisfactory picture quality, a coding word length of between 6 to 8 bits /sample in pulse code modulation (PCM) is necessary.[1-5] At a sampling rate of thrice the chrominance subcarrier frequency, this will lead to a bandwidth of about 80 - 106 Megabits/ second.

Linear transformations were found potentially useful in achieving bandwidth reduction. This reduction is possible due to partial elimination of redundancy inherent in image data. By using a transform, the highly correlated time domain (or spatial domain) data samples are transformed into relatively decorrelated independent coefficients of a series which, if added up again, will restore the original time domain data.

Hotelling[6] firstly derived the transformation that transforms discrete variables into uncorrelated coefficients with a technique which he called ' The Method of Principal Components'. The same principles of Hotelling were used again by Kramer and Mathews[7] , and Huang and Schultheiss[8]. Analogous transformation for continuous data was discovered by Karhunen[9] and Loève[10] and is known as Karhunen-Loève expansion. Kochman[11] and Brown[12] proved that this expansion minimises the mean-squared-truncation error.

After the development of the computational algorithms for the Fast Fourier Transform[13-17], the idea of using this transform for image coding began to draw increasing attention. Investigations for transmitting the two-dimensional Fourier transform of an image rather than the image itself were carried out[18-21] . However, computational difficulty and complexity of hardware realisation of real-time Fourier transform led to the desire in exploiting other kinds of transforms.

One transform which is directly related to Fourier transform is known as Binary Fourier Transform, or BIFORE [22-25]. As this transform is based upon the Hadamard matrix [26] it is known also as 'Hadamard transform'. Other transforms which have been investigated in image processing and are well documented in literature are [27], Haar Transform, Slant Transform [28], and Discrete Cosine Transform [29-31].

Among all these transforms, Hadamard Transform was, and is still, extensively being investigated due to its simple computational algorithms and relatively simple hardware realisation. While Fourier transform produces a decomposition of an image into an infinite series of sine and cosine wave forms, the Hadamard transform produces a finite set of rectangular waveforms.

1.2. Outlines of the work.

The work reported in this Thesis can be broadly divided into two main parts. Tha first part, which comprises the rest of this Chapter, Chapter 2, and Chapter 3, is devoted entirely to the investigation of Hadamard transform and the characteristics of transform domain coefficients. The second part comprises the remaining Chapters, and discusses adaptive coding for different kinds of signals and transforms.

In the rest of Chapter 1, the mathematical fundamentals of Hadamard transform are reviewed. The potential use of the transform in bit rate reduction, based on the general characteristics of transform domain coefficients, is explained. Some existing bit rate reduction techniques are also discussed. These include zonal sampling, threshold sampling, zonal sampling with nonlinear quantization, and the principles of adaptive coding using local characteristics of subpictures.

Chapter 2 studies the characteristics of transform coefficients of monochrome signals in detail. Both one dimensional transform, known also as line transform, and two-dimensional transform, known as block transform, are considered. Statistical studies include probability distribution of coefficients, their variances and their absolute values. As a measure of the transform ability of compacting spatial domain energy in fewer transform coefficients, the energy packing efficiency is also discussed. Different sizes of transform vectors and blocks are cosidered. A general comparison between line and block transforms is presented, aimed at highlighting their relative merits.

Chapter 3 studies the transform domain coefficient characteristics of colour signals. Due to the special character of the composite colour waveform, three different methods of applying the transform are discussed in order to see which one can eliminate most spatial redundancy.

As the degree of correlation among spatial samples is strongly

dependent on sampling frequency, all studies of that Chapter are performed twice,once for the thrice chrominance subcarrier sampling frequency, and again for a four-time subcarrier frequency. A comparison is held at the end of the Chapter between the two sampling frequencies, with particular reference to the suitability of each to the different methods of transform application.

Chapter 4 considers the first problem encountered with the practical use of transformation. This is the inevitable error which results from limiting the hardware word length representing the values of transform coefficients. Levels of noise values for transform sizes and shapes are obtained. Different versions of mean-squared-error criterion are defined for use as an assessment measure. The relation between this criterion and some reported subjective assessment results are presented to link the mere numbers and figures of an objective measure with the subjective assessments of the human eye.

In Chapter 5, monochrome adaptive coding is investigated. A review of some existing adaptive coding techniques is presented with critical comments. The 'activity index' suggested in a previous work to assess the local activities of subpictures in order to select the suitable coder-decoder is critically reviewed. Due to the disadvantages of such an index, the need has arisen for another means of activity assessment. The Chapter, thereafter, is devoted to the establishment and testing of new means of adaptivity selection called the ' Directing Index '. This index is introduced to direct the differnt subpictures to their most suitable coders-decoders. Different indexes are discussed, each with its own computational complexity and advantages.

Coders to be used with each category of subpicture characteristics are devised, based on statistical analyses. Three indexes in all are discussed with the limitations and potentialities of each. Tests for

assessing these indexes are made with an arbitrary average bit rate of 2 bits per pixel. Results are compared with both non-adaptive and existing adaptive schemes for the same bit rate. Although the assessment criterion used is the mean-squared-error, a proposal is made to reduce the stringency of the measure, and to account for the transform character of distributing the errors all over its area.

Chapter 6 discusses adaptive coding of colour signals. Directing indexes devised for monochrome signals are extended for application to colour signals. The applicability of these indexes to the different types of colour transforms is studied.

The sensitivities of both indexes and coders with respect to the sampling frequency used are also discussed. A critical comparison of all the types of transforms discussed is also presented.

Chapter 7 discusses a special coder for coding the high dynamic range low sequency coefficients. Statistical studies of conditional probabilities of coefficients, especially low sequency ones, given that the lowest sequency term is less than a threshold, are presented. Appropriate sets of coders for different cases of values in the lower sequency group are devised. The limits on extending to more coefficients are also discussed, together with the trade-off between complexity involved and advantages gained. Finally an assessment of the coder is presented.

1.3. Hadamard Matrix :

Hadamard matrix is a square matrix whose elements are plus and minus ones only.[32-35] Its rows and columns are orthogonal to each other. Two characteristics of the Hadamard matrix are:

1. Product of the matrix and its transpose is the identity matrix

$$[H]\,[H^t] = N[I] \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.1)$$

Where [ ] denotes a matrix, and t denotes transposition.

[H] is the Hadamard matrix

[I] is the identity matrix

N is the order of the matrix (number of rows or columns).

If [H] is a symmetric Hadamard matrix, then Equation (1.1) reduces to

$$[H]\,[H] = N\,[I] \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.2)$$

2. The Hadamard matrix can be generated recursively as follows:

$$[H_{(k+1)}] = \begin{bmatrix} [H_{(k)}] & [H_{(k)}] \\ [H_{(k)}] & -[H_{(k)}] \end{bmatrix} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.3)$$

where $[H_{(k)}]$ is the Hadamard matrix of order k, and

$[H_{(0)}] = [1]$ is the lowest order matrix. . . . . (1.4)

Hence,

$$[H_{(1)}] = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.5)$$

The simplest construction is for a Hadamard matrix of order N where $N = 2^n$ (n is a positive integer). In such a case, a more useful recursive formula will be to generate a 'double order matrix', as follows.

If [H] is a Hadamard matrix of order N, then a Hadamard matrix of order 2N will be

$$\begin{bmatrix} [H] & [H] \\ [H] & -[H] \end{bmatrix} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.6)$$

Hadamrd matrices of orders 2, 4, and 8 are shown in Figure (1.1).

a. Hadamard matrix of order N= 2.

b. Hadamard matrix of order N= 4.

c. Hadamard matrix of order N= 8.

$$
\begin{array}{cc}
\text{Matrix} & \text{Sequency.} \\
\left[\begin{array}{cc} 1 & 1 \\ 1 & -1 \end{array}\right] & \begin{array}{c} 0 \\ 1 \end{array} \\
\left[\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{array}\right] & \begin{array}{c} 0 \\ 3 \\ 1 \\ 2 \end{array} \\
\left[\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{array}\right] & \begin{array}{c} 0 \\ 7 \\ 3 \\ 4 \\ 1 \\ 6 \\ 2 \\ 5 \end{array}
\end{array}
$$

Figure (1.1). Hadamard matrices of some orders $N= 2^n$.

Along each row of the Hadamard matrix, there is a unique number of sign changes which is a characteristic for this particular row. Harmuth[36] used the term 'Sequency' , rather than 'Frequency', as half the number of zero crossings or sign changes and denoted it by ϕ. As a unit of "sequency", he used 'zps'(one half the average number of zero crossings per second) in analogy to 'cps'. Although several authors and researchers adopted[37] the term 'Sequency', they used it actually as the total number of zero crossings, and not its half.This will be used here as well, i.e. the term sequency will be used as the actual number of zero crossings, or sign changes. Sequency in a transform size of N will then have all integer values from 0 up to N-1. However, for computational simplicity and to avoid using the subscript 0, the term 'order' will be alternatively used. The order of any coefficient(or term) is equal to its sequency+1.

The concept of sequency, or sign changes associated with different rows in the Hadamard matrix, makes it reasonable to think of the rows as being equivalent to rectangular waves with values of ± 1, with subintervals

1/N units. These functions (or waves) are called Walsh functions.[38-43] Walsh functions for the case of $N = 2^3 = 8$, are shown in Figure (1.2). In the figure, functions are shown in the increasing sequency order, which differs from the 'natural' order shown in Figure (1.1), (this was called natural as it is naturally generated by the recursive matrix). The increasing sequency arrangement is similar to the Fourier sine and cosine harmonics. Algorithms were developed[44-49] to obtain these functions in that order which is also called 'Walsh-Hadamard order', or simply, ' Walsh order '.

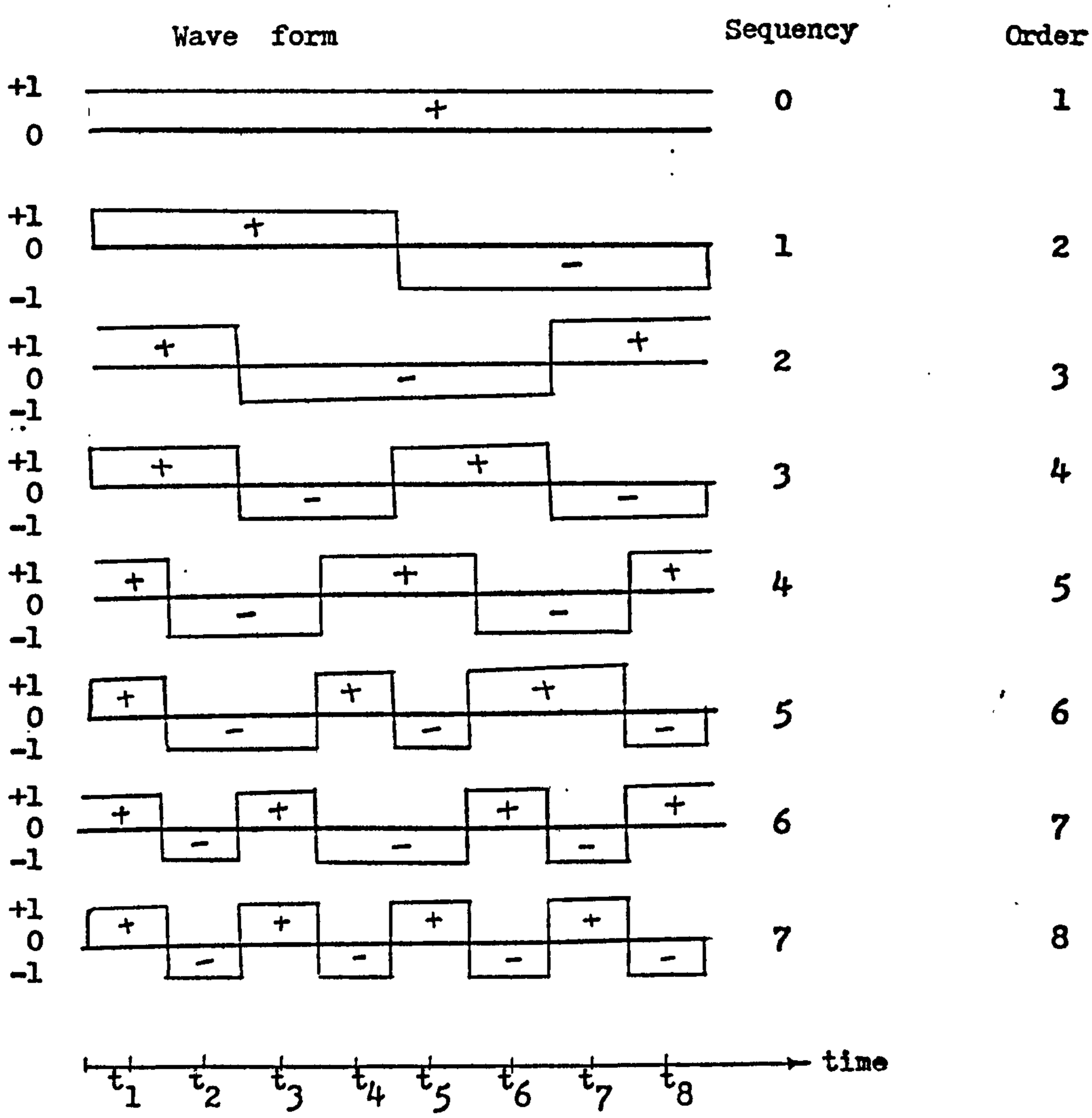


Figure (1.2). Walsh functions for N=8, arranged in increasing sequency order.

1.4. Hadamard Transformation (22,33,37,50-57)

Generally, and for any transform, an original image can be represented by an array of intesity components or samples over the image surface by two-dimensional sampling. This can be done by the usual way of scanning line-by-line, as well as representing each scanned value (sample) by its digitized equivalent value (number). In the transform coding system, it is conceptually possible to transform and process the entire image or sub-sections thereof. For bandwidth compression applications, the choice of the subsection shape as well as size depends heavily on the degree of correlation in the image and on the degree of computational complexity allowed in processing.

As a brief introduction to the transform picture coding, let us assume an image or a section thereof, to be a square array of (N x N) intensity samples described by the spatial function $f_{(x,y)}$ over the coordinates x and y. (Note that the choice of a square array is not a constraint, and that a generally rectangular array can equally be chosen). The two-dimensional transform of the image array $F_{(u,v)}$ defined on a square NxN array of points is expressed as(37)

$$F_{(u,v)} = \sum_{x=1}^{N} \sum_{y=1}^{N} f_{(x,y)} \cdot a_{(x,y,u,v)} \qquad \ldots\ldots\ldots (1.7)$$

where $a_{(x,y,u,v)}$ is the transformation kernel.

In the case of Hadamard transform, the kernel is separable and symmetric,

i.e. $a_{(x,y,u,v)} = a_1(x,u) \cdot a_1(y,v) \qquad \ldots\ldots\ldots\ldots (1.8)$

Hence, the two-dimensional transform can be computed in two steps. In the first, a horizontal transform is taken along every row of the array $f_{(x,y)}$ resulting in the one-dimensional transform:

$$F_{(u,y)} = \sum_{x=1}^{N} f_{(x,y)} \cdot a_{1(x,u)} \qquad \ldots\ldots\ldots\ldots (1.9)$$

Second transform step is a vertical transform which is taken along each column of the 'semi-transformed' array $F_{(u,y)}$, yielding :

$$F_{(u,v)} = \sum_{y=1}^{N} F_{(u,y)} \cdot a_{1(y,v)} \quad \ldots\ldots\ldots (1.10)$$

This is the two-dimensional transformed array.

Assuming no processing on this transformed image, and assuming that the inverse transform kernel corresponding to this transformation is $b_{(x,y,u,v)}$, then, the original image can be restored as :

$$f_{(x,y)} = \sum_{u=1}^{N} \sum_{v=1}^{N} F_{(u,v)} \cdot b_{(x,y,u,v)} \quad \ldots\ldots\ldots (1.11)$$

In matrix notation, let:

[f] be the image matrix in spatial domain (x,y),

[F] be the transformed image matrix (coefficients of transform),

[A] be the transform matrix (the Hadamard matrix in the case of Hadamard transform).

For a transform kernel that is separable symmetric like the Hadamard one, then by matrix multiplication ;

$$[F] = [A] [f] [A] \quad \ldots\ldots\ldots\ldots\ldots (1.12)$$

Assuming an inverse matrix [B] , and by pre- and post- multiplication of each side of Equation (1.12) :

$$[B] [F] [B] = [B] [A] [f] [A] [B] \quad \ldots\ldots\ldots\ldots (1.13)$$

If [B] is chosen in such a way that $[B] = [A]^{-1}$, then Equation (1.13) becomes :

$$[A]^{-1} [F] [A]^{-1} = [A]^{-1}[A] [f] [A] [A]^{-1} \ldots (1.14)$$

But as $[A]^{-1}[A] = [A] [A]^{-1} = [I]$, (the identity matrix), then:

$$[A]^{-1} [F] [A]^{-1} = [I] [f] [I] = [f] \quad \ldots\ldots\ldots\ldots (1.15)$$

i.e. $f_{(x,y)}$ and $F_{(u,v)}$ can be expressed as two-dimensional transform pairs if [A] has an inverse.

The Hadamard matrix is a real, symmetric, unitary matrix known as a symmetric orthogonal matrix . For this kind of matrix :

$$[A]^{-1} = [A] \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.16)$$

Hence, if [H] is a symmetric Hadamard matrix of order N, then :

$$[H]\,[H] = N\,[I] \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.17)$$

Thus, a Hadamard matrix multiplied by a normalizing factor $1/\sqrt{N}$ is an orthonormal matrix.

For symmetric Hadamard matrices of order $N=2^n$, the two-dimensional Hadamard transform may be written in series form (of limited number of terms, or coefficients) as :

$$F_{(u,v)} = \frac{1}{N}\sum_{x=1}^{N}\sum_{y=1}^{N} f_{(x,y)}\,(-1)^{p(x,y,u,v)} \qquad \ldots\ldots\ldots\ldots (1.18)$$

where $$p(x,y,u,v) = \sum_{i=1}^{n}(u_i\,x_i + v_i\,y_i) \qquad \ldots\ldots\ldots\ldots\ldots (1.19)$$

u_i ,v_i ,x_i ,and y_i in Equation (1.19) are the binary forms of u, v ,x ,and y respectively. For example:

$$(u)_{decimal} = (u_n u_{n-1} \ldots\ldots\ldots u_2 u_1)_{binary} \qquad \ldots\ldots (1.20)$$

where $u_i \in \{0,1\}$. The summation in Equation (1.19) is a modulo two addition.

Equation (1.18) gives the Hadamard transform in the so called natural order. Another series representation to obtain the Hadamard transform terms in increasing sequency order form is :

$$F_{(u,v)} = \frac{1}{N}\sum_{x=1}^{N}\sum_{y=1}^{N} f_{(x,y)}\,(-1)^{q(x,y,u,v)} \qquad \ldots\ldots\ldots (1.21)$$

where in this case,

$$q(x,y,u,v) = \sum_{i=1}^{n}\phi_i(u)x_i + \phi_i(v)y_i \qquad \ldots\ldots\ldots\ldots (1.22)$$

and
$$\begin{aligned}
\phi_1(u) &= u_n \\
\phi_2(u) &= u_n + u_{n-1} \\
\phi_3(u) &= u_{n-1} + u_{n-2} \qquad \ldots\ldots\ldots\ldots\ldots\ldots (1.23)\\
&\ldots\ldots\ldots \\
\phi_n(u) &= u_2 + u_1
\end{aligned}$$

and similarly for $\phi_i(v)$ with respect to v .

1.5. Bit Rate Reduction :

The potentiality of linear transformation use in bit rate reduction is based on treating the decorrelated transform coefficients comprising more energy in a way different from those with less energy. In transmitting the transform of an image rather than the image itself, statistical properties of the transform domain have to be exploited in order to reduce the necessary bandwidth. It is then necessary to use a separate quantizing scheme for each coefficient, or group of coefficients. Theoretically, and under ideal coding, the scene and its transform can be transmitted with the same channel capacity. This is true for Fourier transform since it has been shown (58) that the entropy of a scene and its Fourier transform are identical. This result also holds true for the Hadamard transform since its Jacobian is unity.(37)

Quantization is simply the operation of dividing the whole range of the values of the component to be quantized (transform coefficient in this case), into a number of regions separated by quantizing levels (could also be called cut points) (53). All values falling in one region are replaced by a single value (called representative value, or code).

In the selection of transform quantization and reconstruction levels (cut points and representatives), the error criterion depends upon the application of the reconstructed images. For subjective viewing, the relative spatial error for low brightness images provides an indication of image quality. This is based on the fact that incremental brightness changes in the reconstructed image are much more noticeable if the brightness level is low, than if it is high. Thus, to minimise the relative spatial error, the density of quantization levels in the spatial domain should be greater at the lower amplitude levels.

Some existing schemes of quantization in the transform domain in order to reduce the bit rate are now considered.

1.5.1. Zonal Sampling:

As the energy in the transform plane tends to be clustered at certain low sequency coefficients (or terms), the most obvious method of conserving bandwidth is simply to not transmit the high sequency components (or coefficients).

It has been shown[37] that, for some categories of images, about 95 % of the image energy is contained in 1 % or less of the Fourier domain coefficients. As for the Hadamard transform, as the number of coefficients is limited by the transform size, it is to be generally expected that the proportion of Hadamard transform coefficients for the same energy content will be greater than in the case of Fourier transform. Actually, for quite large number of subpicture sizes and for different images, it was found, in this present study, that the first coefficient, H1, contains almost more than 95 % of the total subpicture energy content. But as number of coefficients is limited up to 32 or 64 in a practical transform, this coefficient alone may represent more than 3 % of the total number of coefficients.

Discarding the high spatial sequencies is equivalent in its effect on the reconstructed image to passing the image through a circular zonal low pass filter; the result is a loss of focus. This is due to the fact that the high frequency brightness transitions are important, even though they are relatively few in number, and contain a low proportion of the image energy.

This 'loss of focus' effect can be explained as follows:

Assume a Walsh ordered Hadamard matrix (i.e. one with increasing sequency order) of order 8 for simplicity. The transform of an 8- sample one dimensional array subpicture [S] will be(noting that the 1's will be omitted from the matrix for simplicity) :

$$\begin{bmatrix} + & + & + & + & + & + & + & + \\ + & + & + & + & - & - & - & - \\ + & + & - & - & - & - & + & + \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & - & - & + & - & + & + & - \\ + & - & + & - & - & + & - & + \\ + & - & + & - & + & - & + & - \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \\ S_7 \\ S_8 \end{bmatrix} = \begin{bmatrix} H_1 & H_2 & H_3 & H_4 & H_5 & H_6 & H_7 & H_8 \end{bmatrix} \quad \ldots\ldots (1.24)$$

Assuming, for the moment, that the transform coefficients H's will not be processed in any way, and that the constant N= 8, is only a scaling factor, then the reconstructed vector $[\tilde{S}]$ can be obtained as :

$$[H][H_i] = [\tilde{S}_i] \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.25)$$

where $[H]$ is the same Walsh ordered matrix,

$[H_i]$ is a matrix of transform coefficients,

$[\tilde{S}_i]$ is the reconstructed matrix of samples (values of intensities).

The different values of recovered samples will be as shown below:

$$\left.\begin{aligned} \tilde{S}_1 &= H_1 + H_2 + H_3 + H_4 + H_5 + H_6 + H_7 + H_8 \\ \tilde{S}_2 &= H_1 + H_2 + H_3 + H_4 - H_5 - H_6 - H_7 - H_8 \\ \tilde{S}_3 &= H_1 + H_2 - H_3 - H_4 - H_5 - H_6 + H_7 + H_8 \\ \tilde{S}_4 &= H_1 + H_2 - H_3 - H_4 + H_5 + H_6 - H_7 - H_8 \\ \tilde{S}_5 &= H_1 - H_2 - H_3 + H_4 + H_5 - H_6 - H_7 + H_8 \\ \tilde{S}_6 &= H_1 - H_2 - H_3 + H_4 - H_5 + H_6 + H_7 - H_8 \\ \tilde{S}_7 &= H_1 - H_2 + H_3 - H_4 - H_5 + H_6 - H_7 + H_8 \\ \tilde{S}_8 &= H_1 - H_2 + H_3 - H_4 + H_5 + H_6 - H_7 - H_8 \end{aligned}\right\} \quad \ldots\ldots\ldots (1.26)$$

If now, to achieve a bandwidth reduction ratio of 1:2, the high sequency half of coefficients H_5 — H_8 were put to zero, then from Equations (1.26) above :

$$\left.\begin{aligned} \tilde{S}_1 &= H_1 + H_2 + H_3 + H_4 = \tilde{S}_2 \\ \tilde{S}_3 &= H_1 + H_2 - H_3 - H_4 = \tilde{S}_4 \\ \tilde{S}_5 &= H_1 - H_2 - H_3 + H_4 = \tilde{S}_6 \\ \tilde{S}_7 &= H_1 - H_2 + H_3 - H_4 = \tilde{S}_8 \end{aligned}\right\} \quad \ldots\ldots\ldots\ldots\ldots\ldots (1.27)$$

This could have been noticed also from the Walsh- Hadamard matrix, by letting the right hand half columns to be zeros. As each two successive rows, starting from the first row, differ only in the right half columns, the reconstructed samples at each two consecutive locations will have the same value. Thus :

$$\tilde{S}_i = \tilde{S}_{i+1} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.28)$$

where i= 1,3,5,.....N-1 (in general).

Comparing $\tilde{S}_i$ and $\tilde{S}_{i+1}$ with the corresponding original S_i and S_{i+1}, after substituting values of H_i's, it could be shown that:

$$\tilde{S}_i = \tilde{S}_{i+1} = \frac{S_i + S_{i+1}}{2} \quad \ldots\ldots\ldots\ldots\ldots (1.29)$$

where i= 1,3 ,5, ...N-1.

This will have the same effect on the reconstructed image, as if the original sampling frequency were halved.

If, as another example, coefficients H_3 — H_8 were set to zero to achieve a bandwidth compression ratio of 1:4, it can be easily shown that in such a case :

$$\tilde{S}_i = \tilde{S}_{i+1} = \tilde{S}_{i+2} = \tilde{S}_{i+3} = \frac{\sum_{k=0}^{3} S_{i+k}}{4} \quad \ldots\ldots (1.30)$$

where

i= 1, 5. in this case, or 1,5,9,..... N-3, in general.

Again, the reconstructed image will be as if the original sampling frequency were divided by 4.

Generalizing, it can be said that " if the proportion of coefficients transmitted in zonal sampling is the lowest sequency $(\frac{1}{2})^m$ from the total number of coefficients, where m is an integer, then the recovered data will be such that

$$\tilde{S}_i = \tilde{S}_{i+1} = \tilde{S}_{i+2} = \ldots\ldots = \tilde{S}_{i+(m+1)} = \frac{\sum_{k=0}^{2^m-1} S_{i+k}}{2^m} \quad \ldots\ldots\ldots (1.31)$$

where $i = j.2^m+1$, j= 0,1,2,.....

and the 'equivalent' sampling frequency will be as if

$$f_{equivalent} = \frac{f_{sampling}}{2^m} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (1.32)$$

where $f_{sampling}$ is the original sampling frequency ".

This effect, mentioned above, is the major disadvantage for the simplest bandwidth reduction scheme, namely, the zonal sampling. The more the energy content in any higher sequency coefficient among those discarded, the more distorted and less focussed will be the recovered image processed in this way. In case of Hadamard transformation, degradation in image tends to be more noticeable than for the Fourier transform for the same bandwidth reduction ratio because of the rectangular shape of the Hadamard reconstruction wave form. Human eye is very sensitive to the presence of sharp brightness transitions within an image. All transitions occur within one element in case of Hadamard transform, whereas in case of Fourier transform, the brightness transitions are spread over many elements due to the sinusoidal shape of the wave form.

For that reason, namely the reduction of equivalent sampling frequency and the consequent loss of focus, it is sug ested here that some of the high sequency coefficients should be always transmitted. This will avoid the result that several consecutive points in the recovered image will have the same values.

1.5.2. Threshold Sampling : [37,59]

This is the simplest type of ' adaptive ' bit rate reduction schemes. While the major source of degradation in zonal sampling scheme was that large magnitude high sequency coefficients (when exist in the transform domain) were discarded altogether, only the coefficients which are below a threshold value will be discarded in this scheme. Depending on the value of the threshold, some coefficients will be transmitted, and others will be set to zero in the reconstruction process. In recovering the original data, coefficients which were transmitted must be identified in order to be positioned in their right places in the inverse matrix. So, it is necessary to code also the positions of transmitted coefficients, as well as their values. Position coding adds to the transmission bandwidth and bit rate. Experiments[37] show that a threshold value which allows nearly one sixth of the transform coefficients (for particular image category, namely the portrait) provided good quality reconstruction.

1.5.3. Zonal Sampling with Non-linear Quantization

A different procedure can be adopted which is based on quantizing successive transform coefficients more and more coarsely, while discarding some of the high sequency coefficients. Landau and Slepian[53] adopted this tachnique for two reasons :

Firstly, since the lower sequency coefficients have more variability (mean, variance, and standard deviation), as seen in their study, then reproducing these more accurately helps reducing the mean square error for the more frequently occuring picture areas.

Secondly, the higher sequency coefficients tend to be large mainly when the local area has a very 'busy' or choatic nature. The detail of choas is guessed to be less important to the viewer than the existence of the choas itself. Thus, the fidelity criterion behind the scheme contains an element of the characteristics of observers in addition to considerations of mean square error.

In their experiments, a block transformation was considered (of a size 4x4). Number of quantization levels given to each coefficient of the first ten was approximately proportional to the variance of the coefficient, while the last (highest sequency) six coefficients (H_{11} -- H_{16}) were dropped completely. First coefficient, H_1, was quantized by a 64-level uniform quantizer. Coefficients H_2 -- H_{10} were quantized with quantizers having a companding characteristics given by a function of the form $y = k\sqrt{x}$, coinciding with Max type coders[60]. However, the actual cut-points and representative values were varied from the corresponding Max type's values by try and error on subjective assessment basis. Table (1.1) shows the relative variances of coefficients (normalized to the variance of the first term H_1). Table (1.2) shows the number of quantizing levels for each coefficient. Table (1.3) shows the values of cut points and representatives reached after over 100 experiments, as reported.

As a conclusion of that work, and from Table (1.2), an average bit rate of 2 bits/picture sample (pixel) was found to be satisfactory for the used picture category, namely portrait as the case with Picturephone® images used in that experiment.

Table (1.1). Relative variances of different coefficients.

Coeff. order.	Relative variance.	Ceff. order.	Relative variance.
1	1.000	9	0.024
2	0.098	10	0.024
3	0.087	11	0.020
4	0.035	12	0.022
5	0.038	13	0.019
6	0.051	14	0.015
7	0.048	15	0.016
8	0.034	16	0.014

Table (1.2). Number of quantization levels and coding bits for each transform coefficient.

Coeff. order.	Quant. levels.	Coding bits.	Coeff. order.	Quant. levels.	Coding bits.
1	64	6	9	4	2
2	16	4	10	4	2
3	16	4	11	0	0
4	8	3	12	0	0
5	8	3	13	0	0
6	8	3	14	0	0
7	8	3	15	0	0
8	4	2	16	0	0

Table (1.3). Cut points and representative values for a transform block size of 4x4.

Possible values of H_i lie in the range $\pm$ 2048.0, and are integer multiples of 0.25.	
H_1	The range between -1940 and +1900 was divided into 64 intervals of length 60, the mid point of each serving as representative value for the interval. The first and last points also represented any values of H_1 outside this range.
H_2 and H_3	Cut points $\pm$ 0 8.6 34.4 77.4 137.5 215 309 421 Representatives $\pm$ 2.1 19.3 53.7 105 174 260 363 483
H_4 , H_5 H_6, and H_7	Cut points $\pm$ 0 15.6 62.4 140.4 Representatives $\pm$ 3.9 35.2 97.6 191
H_8, H_9, and H_{10}	Cut points $\pm$ 0 62.5 Representatives $\pm$ 15.6
H_{11} --- H_{16}	Dropped altogether.

1.5.4. Adaptive Coding for Different Local Characteristics :

In many transform coding studies,[31,57,61] it was proposed that adaptive systems could be used for bit rate reduction. In a particular reference to Hadamard transform [57], it was suggested that such a transform coding system has to provide sufficient capacity for a wide range of signals and, because coding requirements vary considerably from one type of signals to another, there are always several bits which are not required for coding any one set of transform coefficients. Hence, further reduction of bit-rate could be made by using a form of coding that would omit these unused bits. As the characteristics of transform domain will vary from one subpicture to another, so the coding system would have to adapt its characteristics to the current requirements of the signal. Such a system would have different coders for different subpicture structures. The encoded signal contains two components, one being a code used to inform the receiving equipment of the method used to encode the signal (i.e. the coder), and the other being the signal itself (i.e. transfrom coefficients). It is the main target of this study to devise an objective computable means for assigning such an ' informing ' code to each individual subpicture.

The problem of objectively measuring the activity of a picture has not received adequate attention so far. Gimlett[62] however, proposed an ' Activity Index ' to help assigning appropriate number of bits for coding different coefficients. The probability function of such an index is to be found for a large number of images. Four categories of subpicture activity, for example, are defined by the 25 % segments of the cumulative function. Different quantizing schemes are then to be devised for these categories, depending on the relative values of coefficients. However, it was pointed out that, this was an entirely illustrative example and no attempt has been made to minimise mean-squared-error, or any other subjective error criterion. The scheme has not even been tested.

In this study, a new ' Directing Index ' is established. This index is intended to direct each subpicture to the most adaptive coder in a " stored coder " or " multi-coder " scheme. It has been called " Directing Index ", to emphasize that it is not intended to assess the overall activity of a subpicture,(although it does so implicitly), but to assess the internal relations among different coefficients, or groups of coefficients, within the subpicture transform. In setting this index, advantages were exploited from statistical properties in the transform domain. Coders to be used for different characteristics of local activities are also devised.

CHAPTER TWO

TRANSFORM DOMAIN CHARACTERISTICS OF MONOCHROME SIGNALS

A better understanding of the transform domain characteristics will, undoubtedly, lead to a better exploitation of the benefits of the transform. This Chapter is, therefore, devoted to a study of these characteristics in monochrome television signals. Statistical characteristics of coefficients are studied in order to help in allocating appropriate coding schemes. Energy packing efficiency is investigated to help decide which particular coefficients or groups of coefficients will better be coded.

The two different types of transforms, namely line- and block-transforms are considered. Different transform sizes, as well as different block shapes are investigated.

2.0. Introduction :

The analyses in this Chapter were performed on stationary images. Monochrome photographic slides have been previously coded into 8-bits per sample pulse code modulated (PCM) digital video using a high speed analogue to digital converter (ADC), and stored on paper tapes [63]. A standard sampling frequency of thrice the chrominance subcarrier frequency (= 13.3 MHz) was used for coding of all image signals. However, due to the limited length of paper tape, only the middle 72 lines of any television frame data could be punched and stored.[63] These lines, being in the middle, almost preserve the main features and characteristics of their corresponding frames.

Four test images were used. These are defined and named as follows:

A: VILLAGE.

B: PORTRAIT.

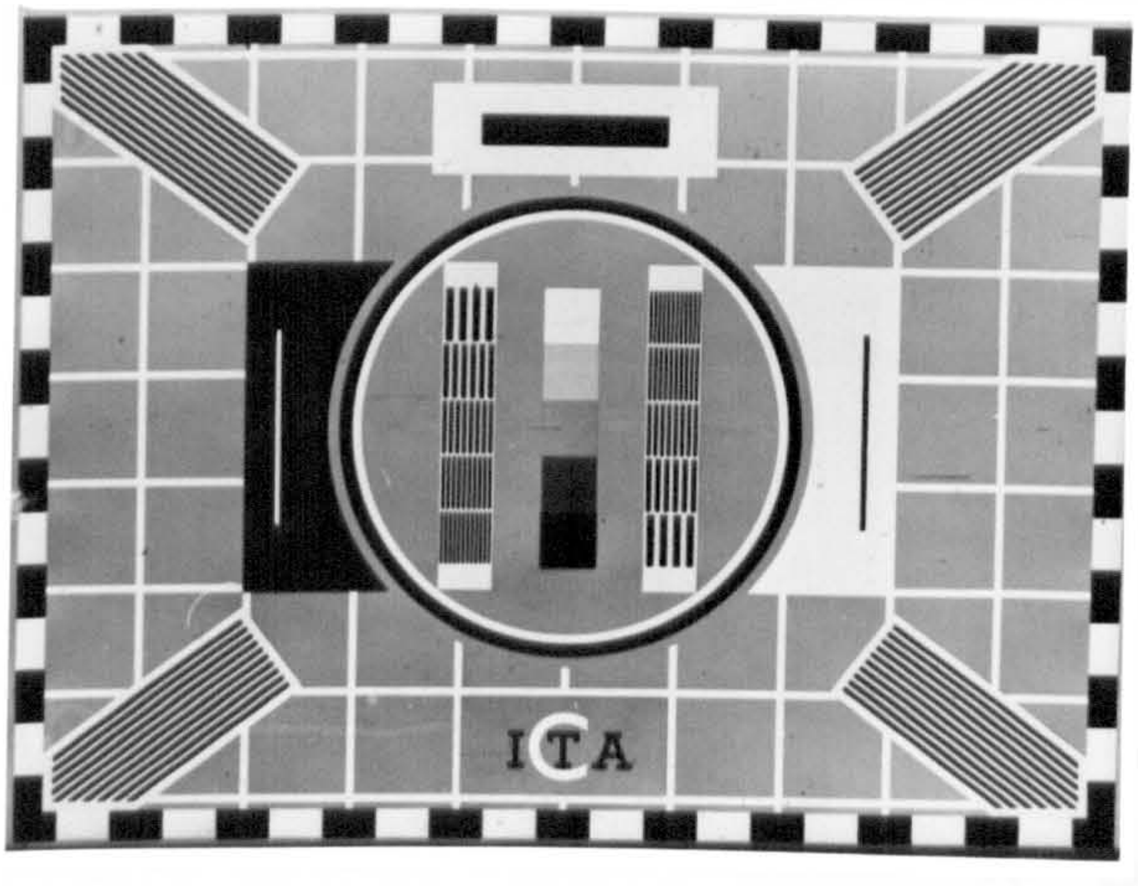
C: TEST CARD.

D: SHIBDEN HALL.

Figure (2.1) shows reproductions of these images.

(a) Image A, Village.

(b) Image B, Portrait.



(c) Image C, Test card.

(d) Image D, Shibden hall.

Figure (2.1). Test images for monochrome analyses.

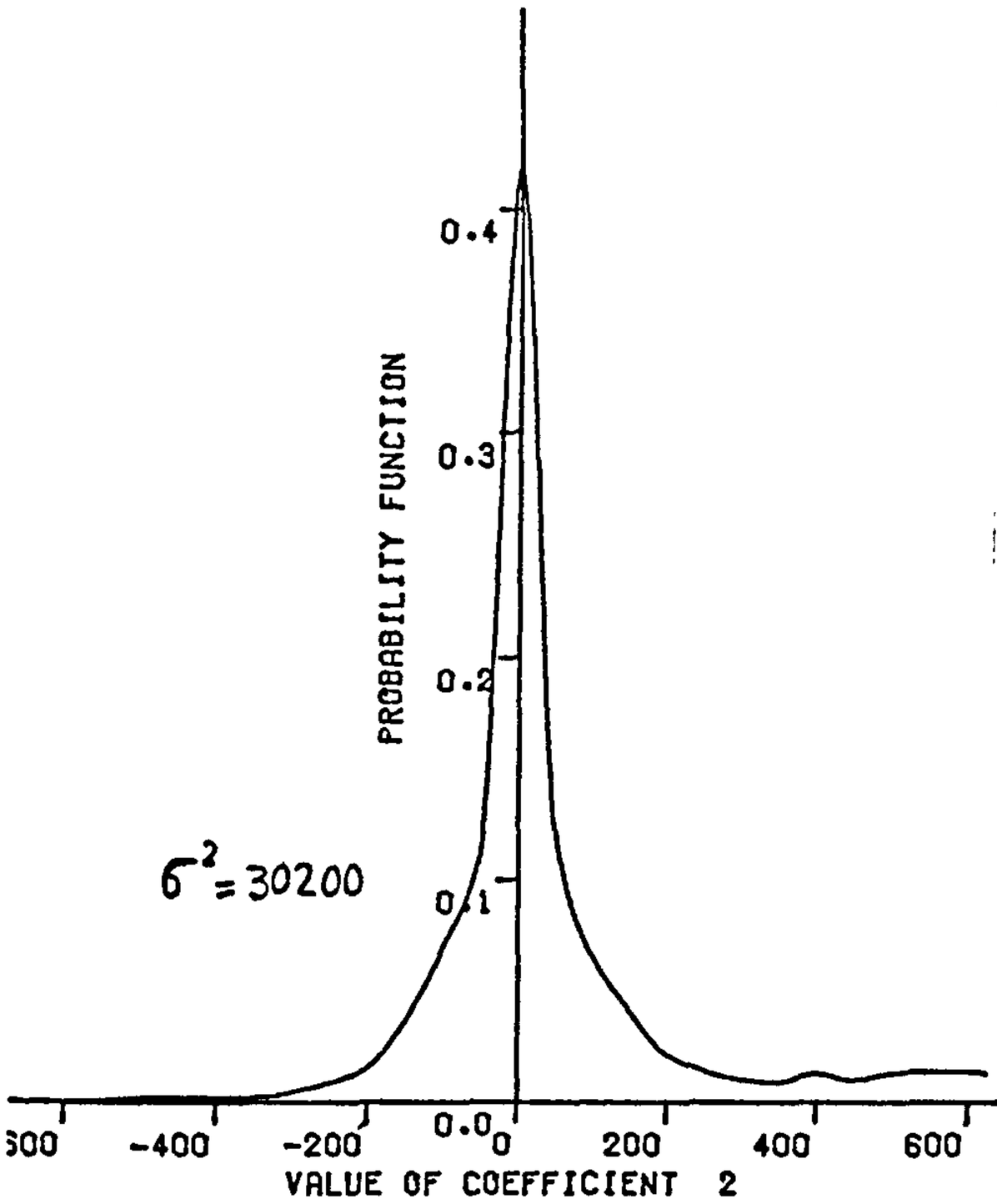
2.1. Line Transformation

In line transformation, which is a one dimensional transformation, the television line samples (pels) are grouped into "Vectors" . For simplicity, number of samples in one vector (N) is chosen to be an integer power (n) of 2, i.e. 4,8,16,32,...etc. The original TV-line samples number was restricted to 640 samples of the active portion of the line, to facilitate grouping each line in an integer number of vectors of size up to 128 each. These samples are then grouped into vectors of the desired sizes. Different sizes have been tested, ranging from N=4 to N= 128,(i.e. from n=2, to n=7). In each case, the vector of samples, considered as a column matrix, was pre-multiplied by a Hadamard matrix of order N and so, the Hadamard transform for a particular vector was obtained.

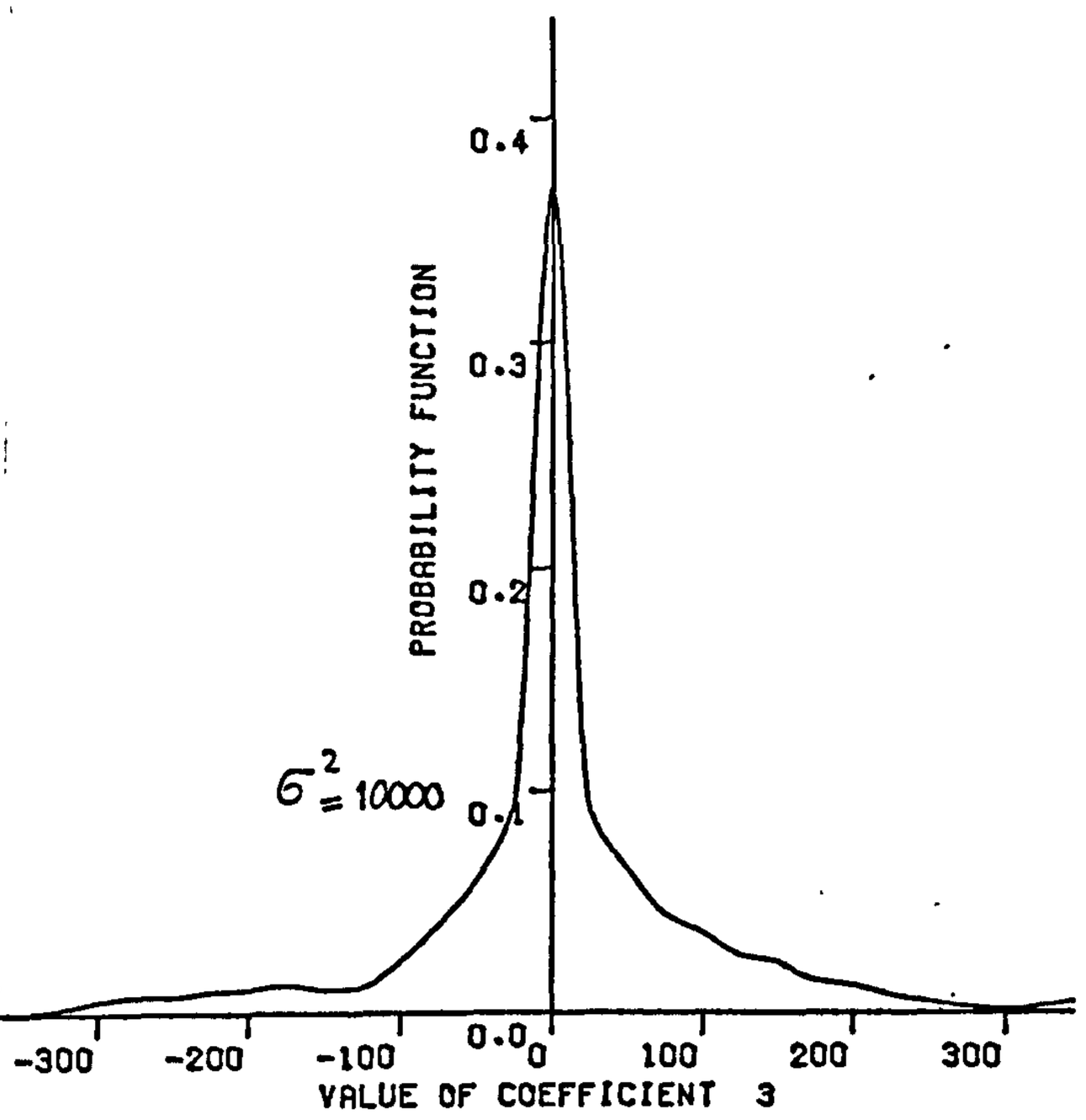
2.1.1. Probability Distribution of Coefficients :

Figure(2.2.) shows the probability density functions ,(P.D.F.), for some coefficients in different image categories, and a transform size of N=16. Although the distribution functions are not exactly normal, only the normal distribution can be thought of as the closest fit especially for low sequency coefficients. For higher sequencies, however, deviations from the approximate normal distributions are greater, and differences between image categories are also widely spread.

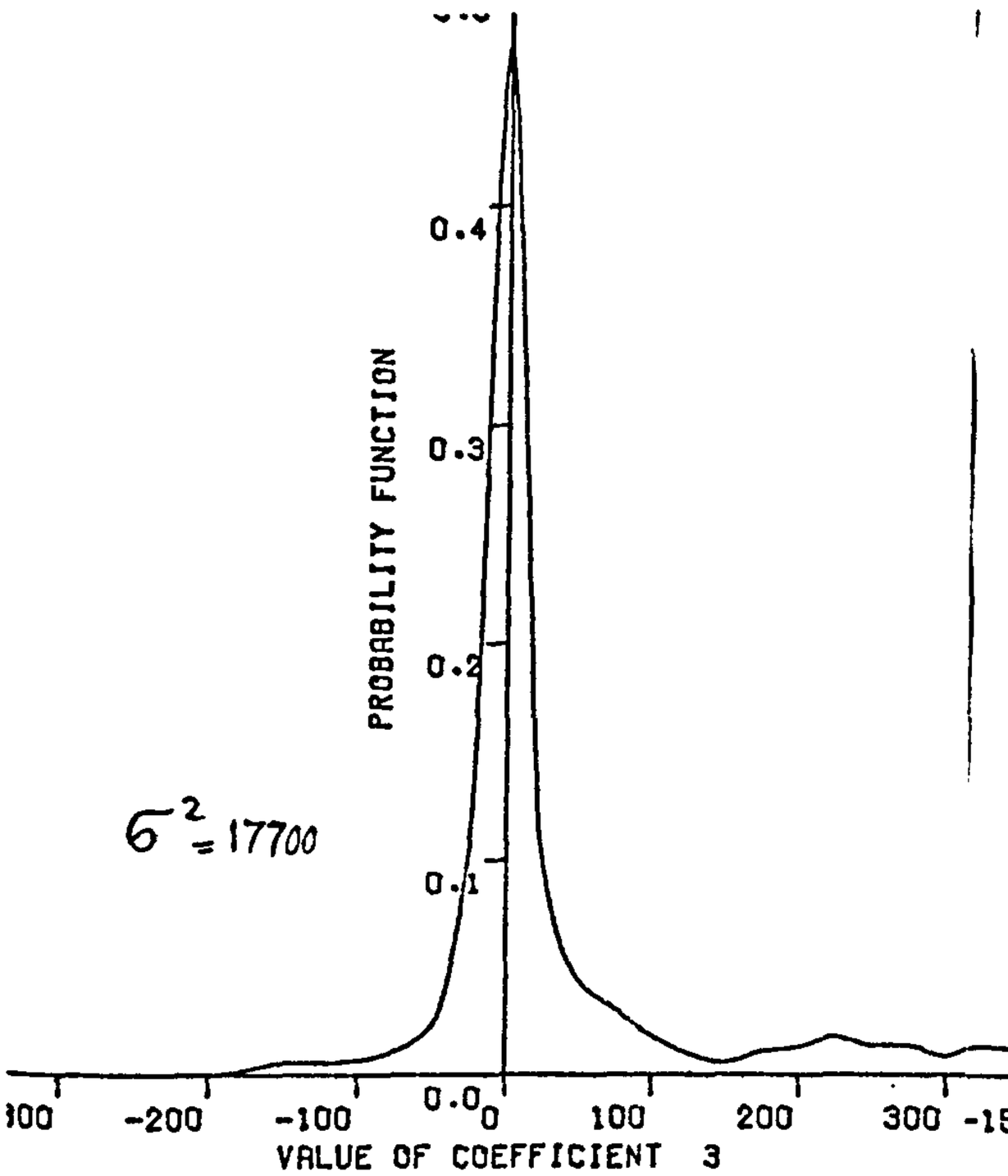
Figure (2.3.) shows the PDF for some coefficients in a transform size of N=16 for a "Hypothetical" image composed by 'mixing' all the four images. The figure shows that a normal distribution is only applicable if a large number of different images are considered. This means that for a particular image category, normal distribution of transform coefficients will be much doubted.



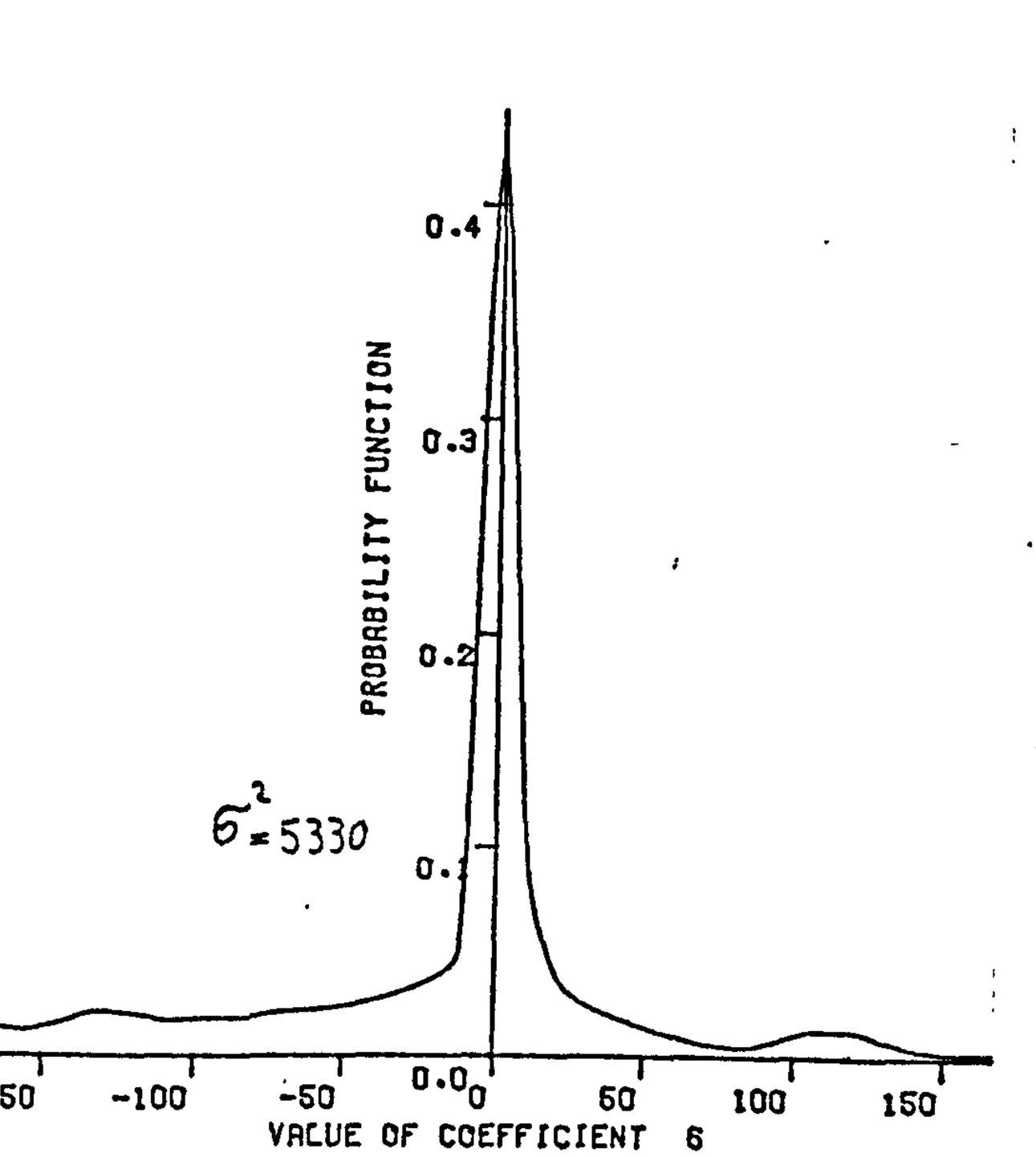
(a) H2, image A.



(b) H3, image A.



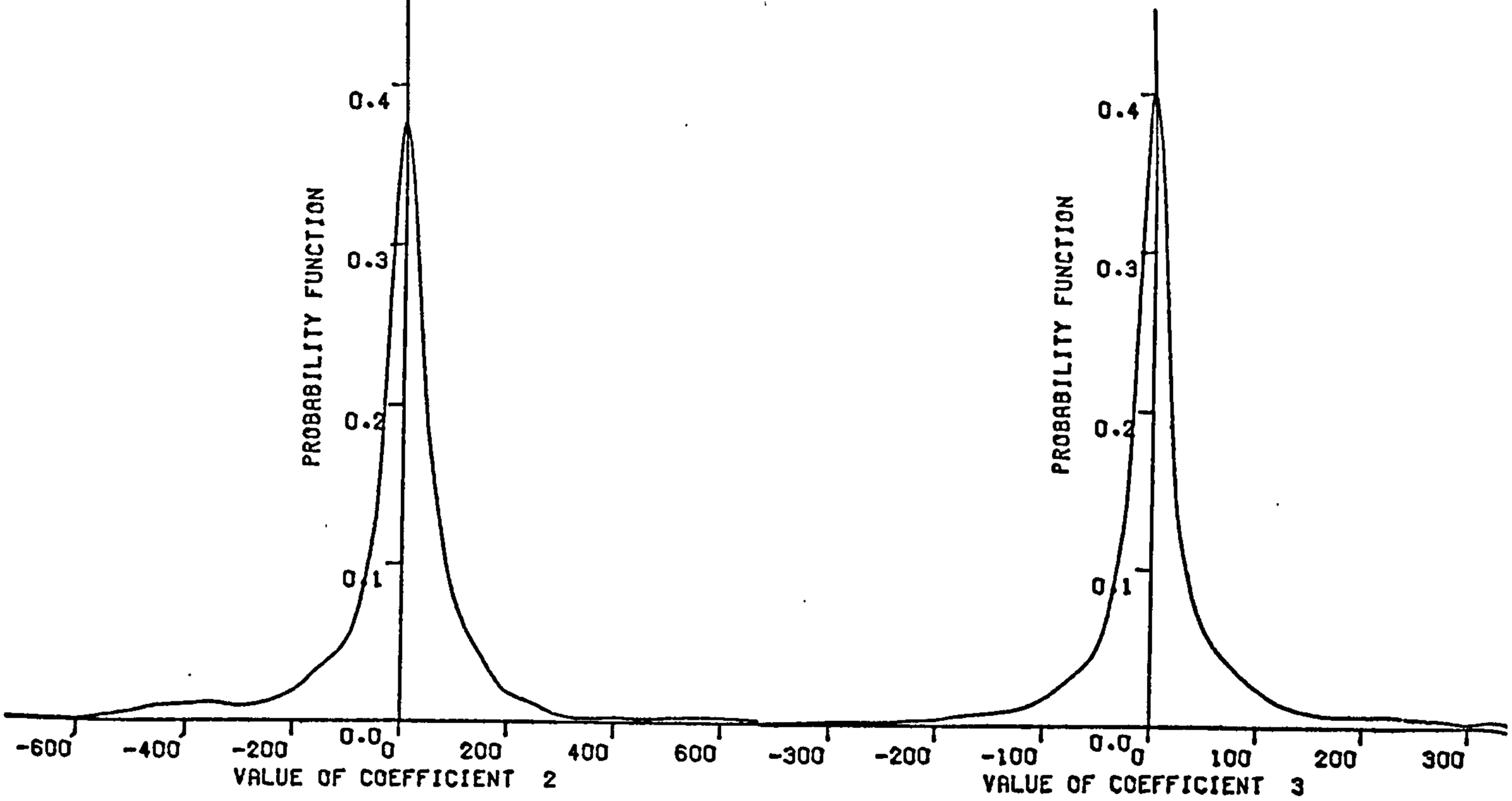
(c) H3, image C.



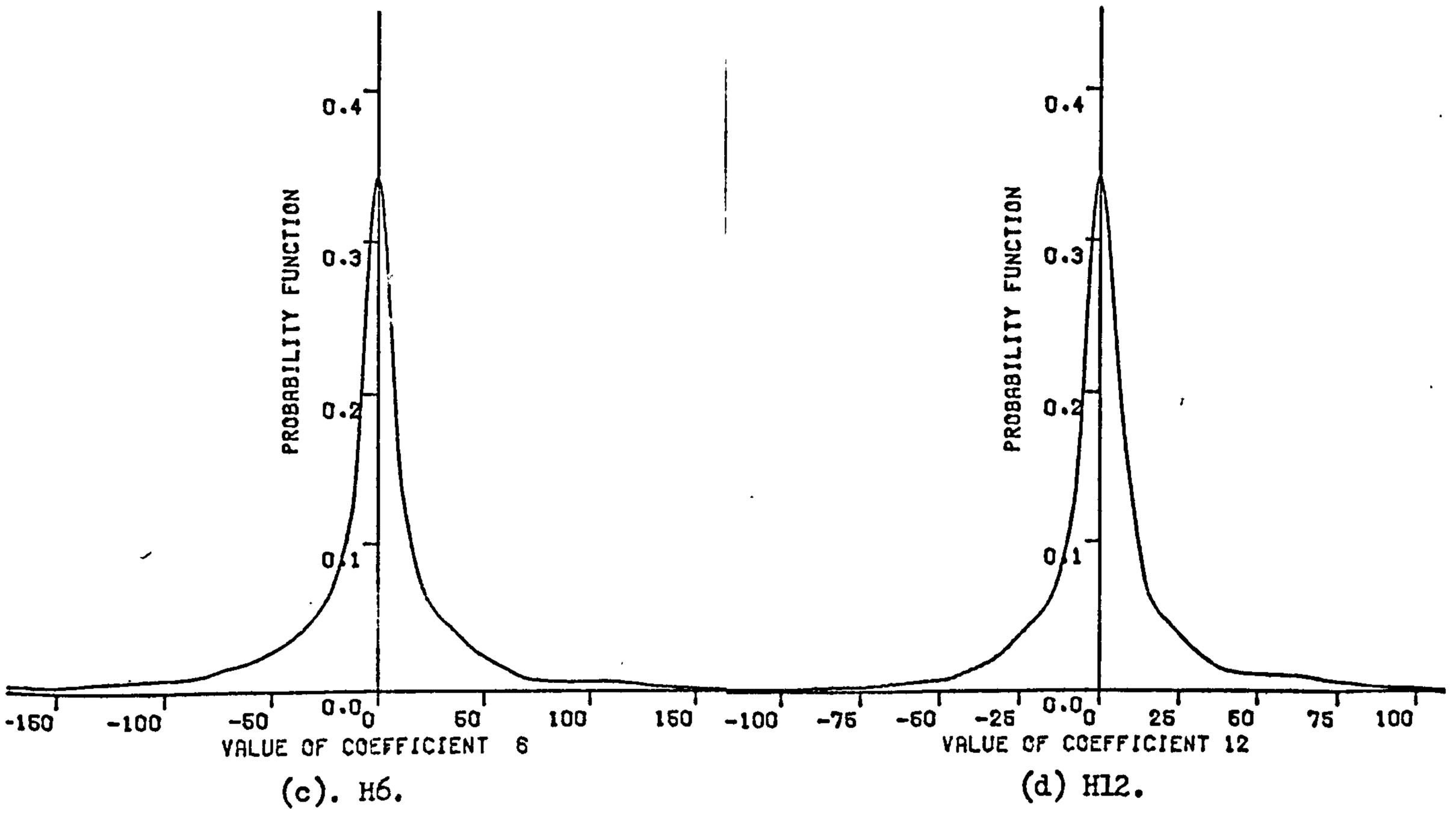
(d) H6, image C.

Figure (2.2). PDF of some coefficients in some images.

σ^2= Variance.



(a) H2. (b) H3.



(c). H6. (d) H12.

Figure (2.3). PDF of some coefficients in a transformed 'hypothetical' average image.

2.1.2. Variances of Coefficients :

Although the general shape of the PDF for most of the ac coefficients, in transform domain, and for all image categories shown is approximately a Gaussian distribution, the variances are not likely equal. Values of variances for each coefficient of those studied earlier in the previous section are shown on corresponding graphs of probability distribution functions. In addition, Tables (2.1 - 2.3) show the values of variances for different transform sizes and different image categories. The tables also show the relative variance for each ac coefficient as compared with the variance of first coefficient H_1 (dc coefficient). This relative variance, expressed in dB for simple comparison, gives a rough idea about the reduction in number of bits required to code a particular coefficient, than those required for the dc term. (31,53) Although expressing this value in dB has no meaning, as there is no power involved, it is used here only as a way of logarithmic scaling of ratios. Another comparison measure is also shown in the tables. This is the ratio between maximum and minimum values of variances of a particular coefficient in different image categories. This measure indicates to what extent the values of variance do depend on the picture category. It can be seen from the tables that a common and a general model for values of variances for all or most of picture categories seems far from reaching. Wide-spread variations in values of variances for a particular coefficient among different pictures are noted, sometimes up to 2870 %. However, a general trend in all cases, practically regardless of the picture category, or the transform size, is the decrease of values of variances as the order (or sequency) of coefficient increases. A careful investigation however, shows that this decrease is not quite uniform or continuous function of sequency. Values of variances of some consecutive coefficients tend to be relatively constant, then decrease considerably for another group of higher sequency coefficients, where again they are relatively constant at this new low value, and so on.

Table (2.1). Variances of different coefficients for different image categories. Transform size N= 4.

Category.	A		B		C		D		Ratio of Maximum to minimum.
Coeff. order.	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	
1	30300	0	19200	0	19700	0	36800	0	1.9
2	726	-32	109	-45	1210	-24	589	-36	11.1
3	140	-47	18	-61	354	-35	144	-48	19.7
4	132	-47	26	-57	240	-38	120	-50	9.2

Table (2.2). Variances of different coefficients for different image categories. Transform size N= 8.

Category.	A		B		C		D		Ratio of Maximum to minimum.
Coeff. order.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	
1	115000	0	75600	0	76000	0	141000	0	1.87
2	5030	-27	1030	-37	9030	-19	5400	-28	8.77
3	1740	-36	204	-51	2170	-31	1400	-40	10.6
4	1090	-40	220	-51	2140	-31	1280	-41	9.7
5	237	-54	34	-67	780	-40	311	-53	22.9
6	308	-51	43	-65	592	-42	313	-53	13.8
7	333	-51	46	-64	506	-44	293	-54	11.0
8	204	-55	53	-63	380	-46	264	-55	7.2

Table (2.3). Variances of different coefficients for different image categories. Transform size N= 16.

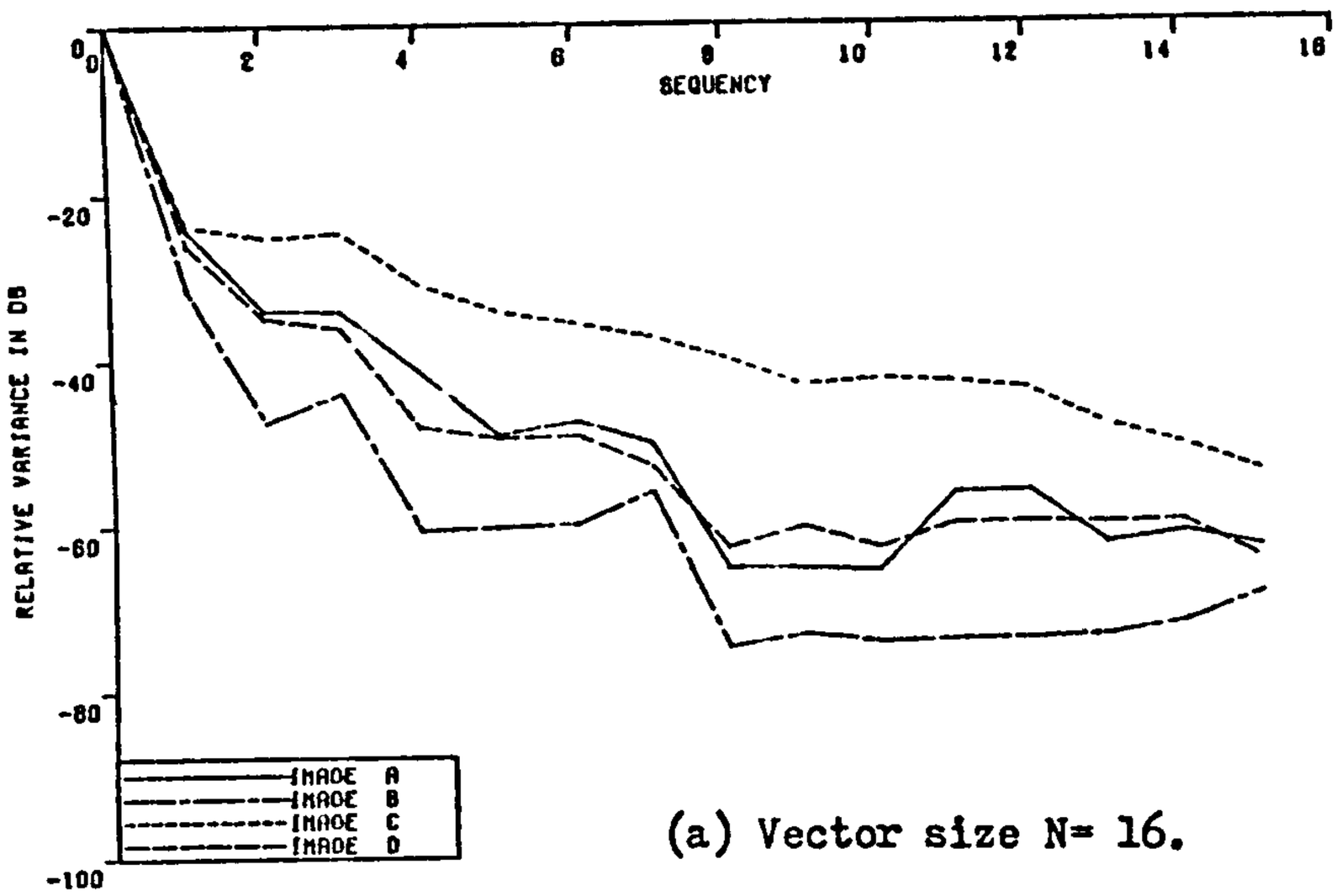
Category.	A		B		C		D		Ratio of
Coeff. order.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	Variance.	Relative var.in dB.	Maximum to minimum.
1	426000	0	290000	0	289000	0	531000	0	1.8
2	30200	-23	11100	-28	37600	-18	45100	-21	4.1
3	10000	-33	2400	-42	17700	-24	10500	-34	7.4
4	9260	-33	2700	-41	10800	-29	10200	-34	4.0
5	4090	-40	375	-58	4630	-36	2650	-46	12.3
6	2450	-45	458	-56	5330	-35	2900	-45	11.6
7	2380	-45	472	-56	4350	-36	2690	-46	9.2
8	1950	-47	597	-54	3290	-39	2350	-47	5.5
9	426	-60	69	-72	1980	-43	617	-59	28.7
10	432	-60	75	-72	1240	-47	640	-58	16.5
11	434	-60	89	-70	1350	-47	696	-58	15.2
12	737	-55	89	-70	974	-49	637	-58	10.9
13	775	-55	89	-70	1110	-48	578	-59	12.5
14	502	-59	102	-69	1130	-48	638	-58	11.1
15	485	-59	115	-68	831	-51	555	-60	7.2
16	359	-61	144	-66	482	-56	511	-60	3.5

As an illustrative example, from Table (2.3), for a vector size of N= 16, a comparison shown in Table (2.4) is derived.

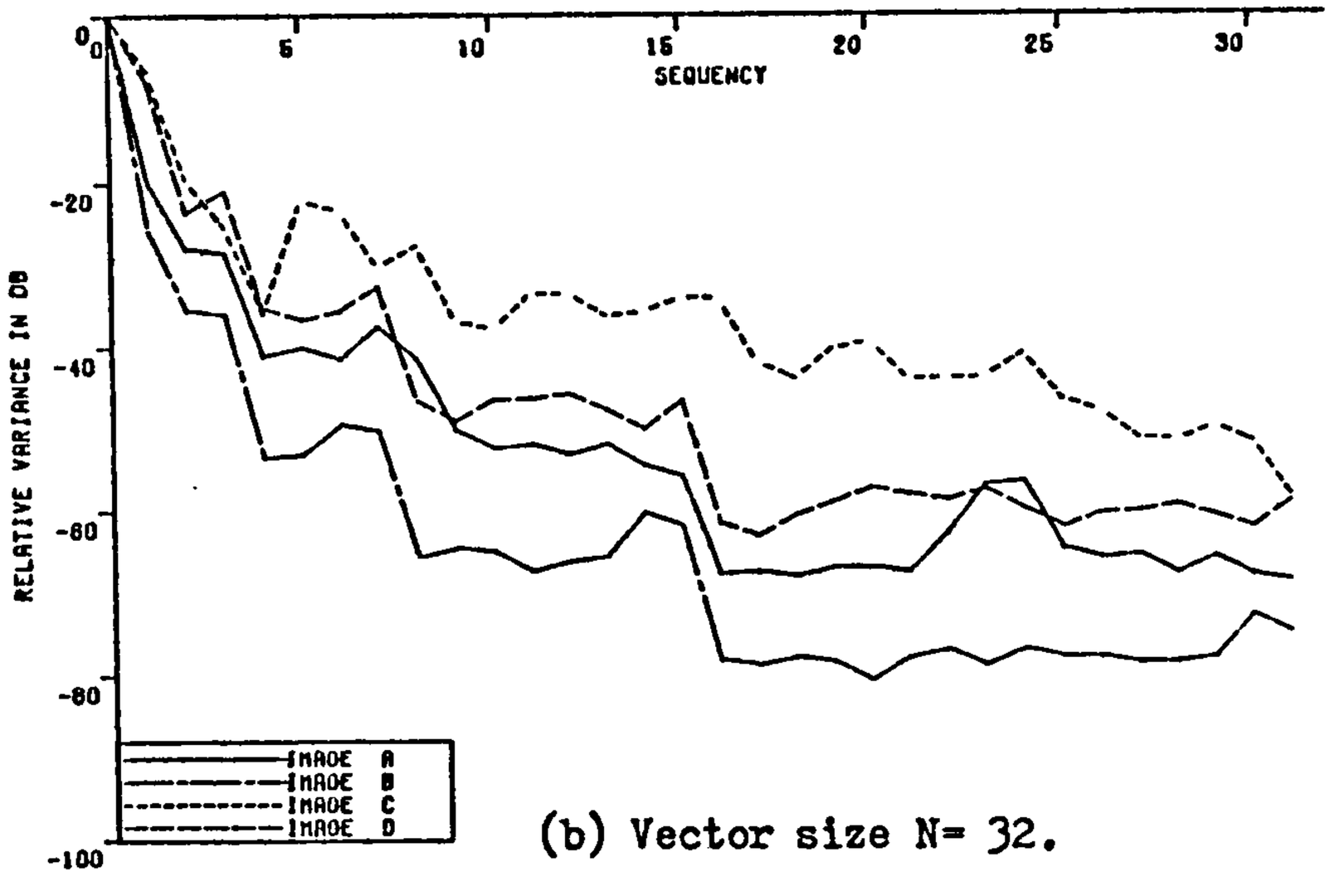
Table (2.4). Ranges of values of relative variances for different groups of coefficients.
(Note: minus signs were dropped for clarity).

Coefficients orders ; (sequency + 1)	Ranges of values of relative variances in dB.			
	A	B	C	D
2	23	28	18	21
3-4	33	41-42	24-29	34
5-8	40-47	54-58	35-39	45-47
9- 16	55-61	66-72	43-56	58-60

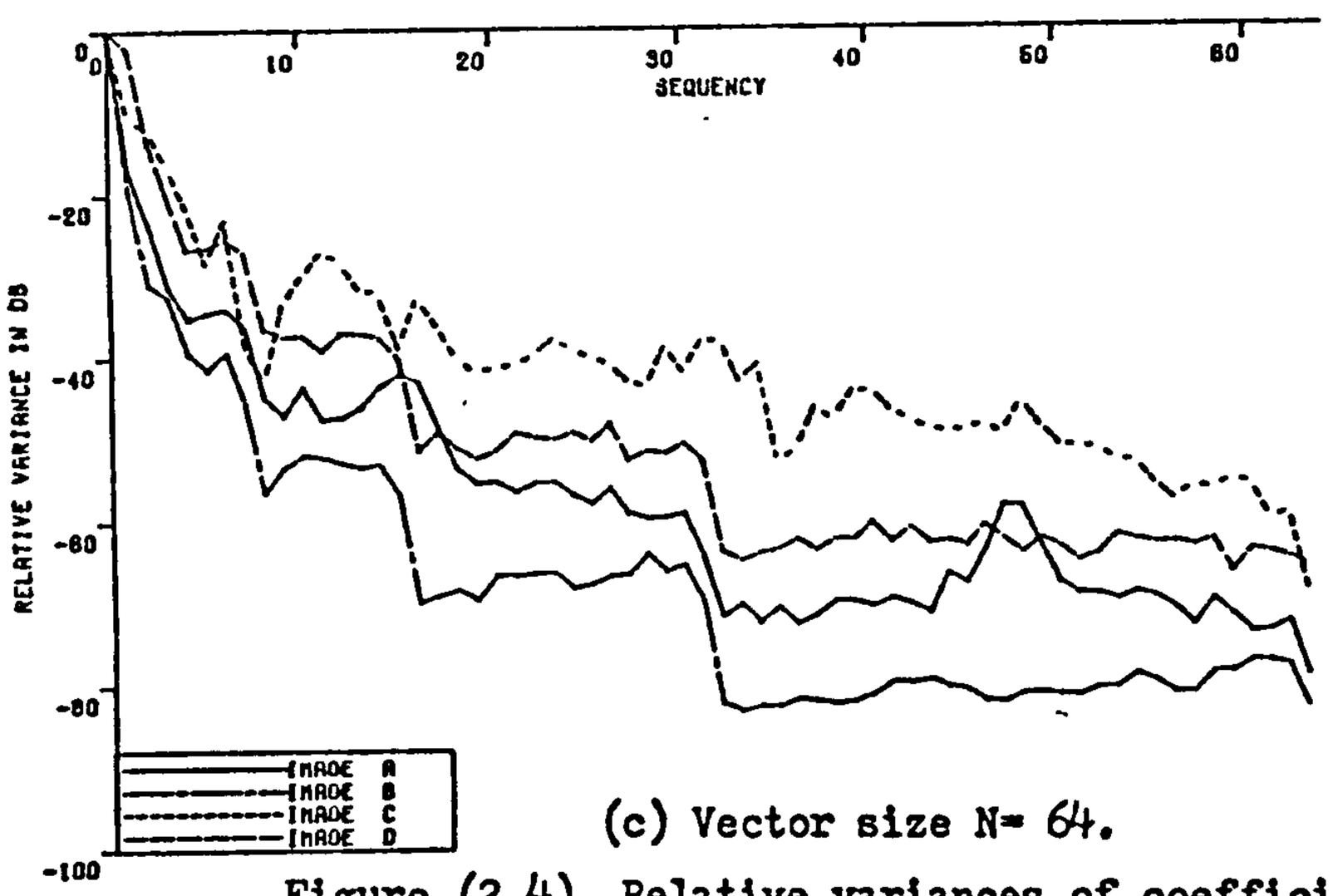
This means that , generally,coefficients in a particular group have nearly the same variances. As the order of the group, rather than the order of coefficients, increases, coefficients are more quiet and have less spread values among the statistical mean of nearly zero. Graphs for remaining transform sizes are shown in Figures (2.4).



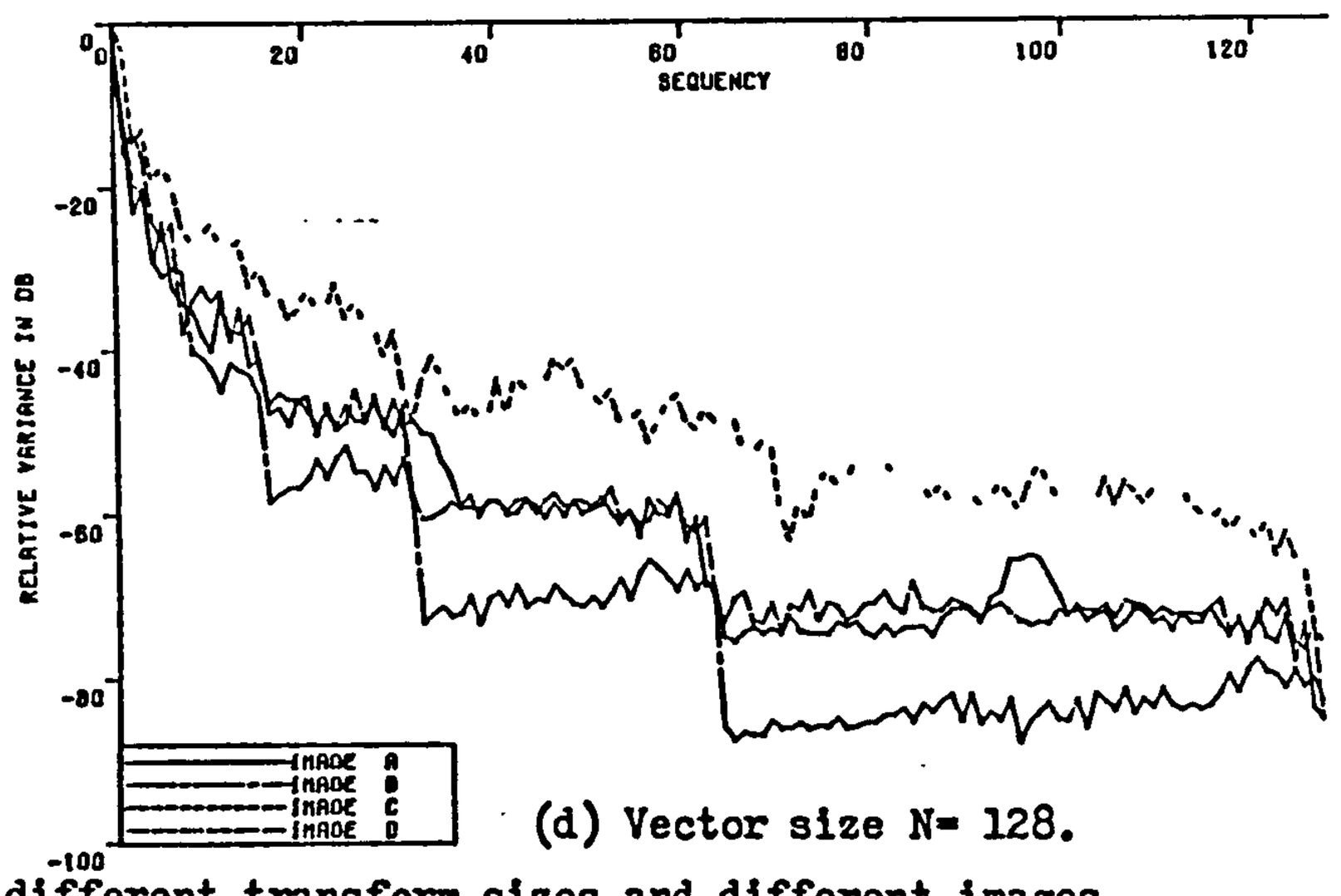
(a) Vector size N= 16.



(b) Vector size N= 32.



(c) Vector size N= 64.



(d) Vector size N= 128.

Figure (2.4). Relative variances of coefficients in different transform sizes and different images.

2.1.3. Absolute Values of Transform Coefficients :

Another statistical variable which is of great importance in processing transform coefficients is the absolute value of each coefficient.

As has been seen from the definition of Hadamard matrix and Transform, values of all coefficients, except the first (D.C. term), may likely be positive or negative. In fact this is why the mean of any particular coefficient , of sequency 1 or more, is zero, or more accurately, close to zero. In coding the coefficients, we are interested in their absolute values.

Extensive statistical studies have been done involving the absolute values of coefficients, for different transform (vector) sizes and different pictures categories. The averages of these absolute values were computed for vector sizes ranging from 4 to 128 samples (i.e. $n = 2$ to 7). As the value of the first (D.C.) coefficient is a measure of the overall luminance (brightness) of the transformed area, it may be more meaningful to express the absolute value of an ac coefficient as a ratio to that dc term. This relative value is then taken as a measure of the role which that particular coefficient will play in recovering the 'original' spatial data.

Tables (2.5 - 2.7) show the averages of absolute values of coefficients for transform sizes from 4 to 16 and for the four image categories. As the order of a coefficient, and hence its sequency, increases, the coefficient value decreases as expected. As the decrease in values may sometimes be very sharp, a logarithmic scale is used to express the relative values of different ac coefficients as compared with the dc term.

Graphs for cases where transform size is bigger than 16 are shown in Figures (2.5).

Table (2.5). Averages of absolute values of coefficients for different image categories. Transform size N= 4.

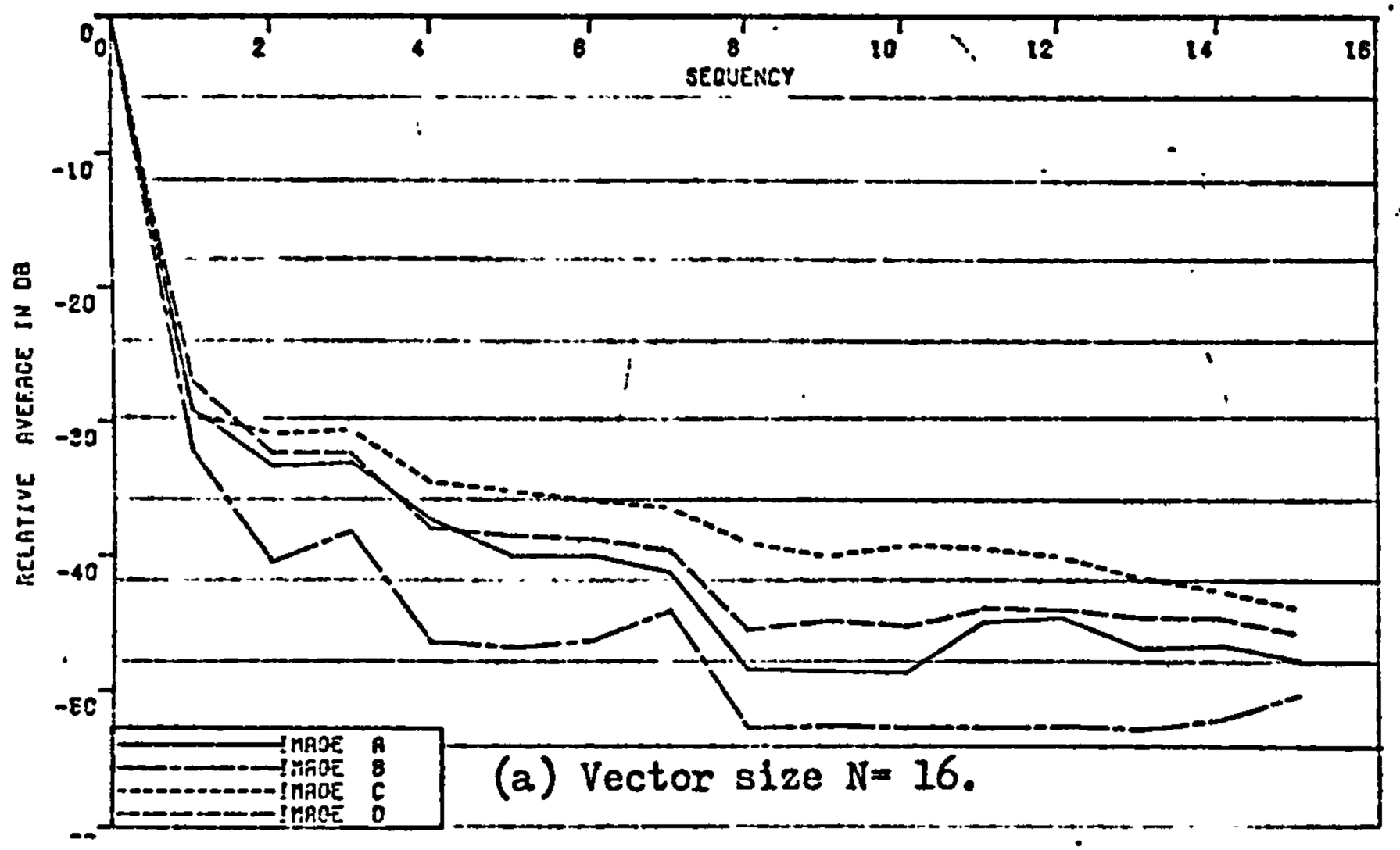
Image categ.	A		B		C		D		Ratio
Coeff. order.	Average.	Relative average, dB.	Average.	Relative average, dB.	Average.	Relative average,dB.	Average.	Relative average, dB.	Maximum to minimum.
1	716	0	657	0	719	0	554	0	1.3
2	15.5	-33	6.0	-41	15.6	-33	13.9	-32	2.6
3	7.0	-40	2.7	-48	8.2	-39	7.1	-38	3.0
4	6.9	-40	3.1	-46	7.5	-40	6.7	-38	2.4

Table (2.6). Averages of absolute values of coefficiennts for different image categories. Transform size N= 8.

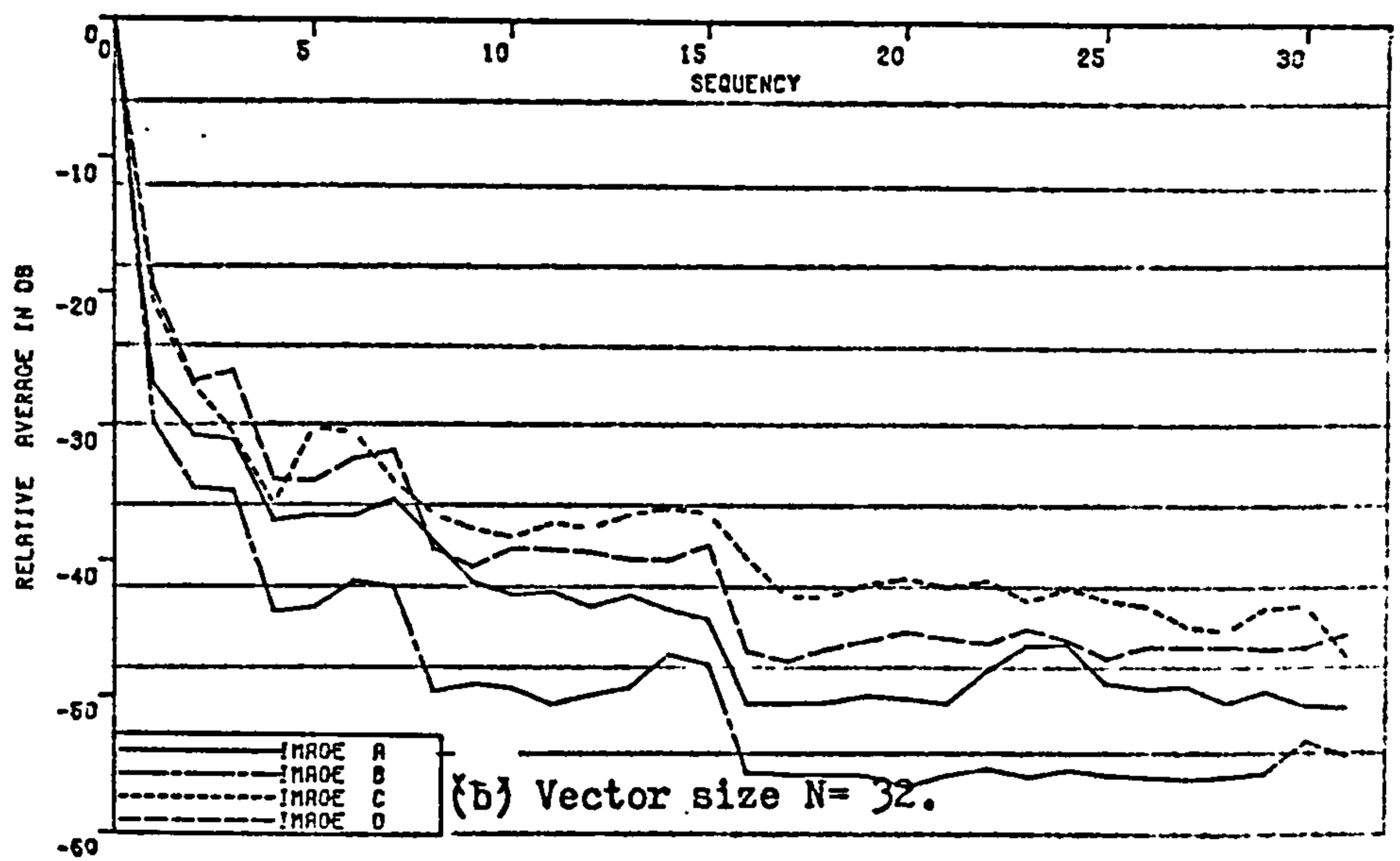
Image categ.	A		B		C		D		Ratio
Coeff. order.	Average.	Relative average, dB.	Average.	Relative average, dB.	Average.	Relative average, dB.	Average.	Relative average, dB.	Maximum to minimum.
1	1430	0	1310	0	1430	0	1100	0	1.3
2	41.2	-31	18.3	-37	44	-30	43.7	-28	2.4
3	24.8	-35	8.1	-44	21.8	-36	22.7	-34	3.1
4	19.7	-37	9.2	-43	23.7	-36	21.6	-34	2.6
5	9.1	-44	3.9	-51	11.5	-42	10.5	-40	2.9
6	10.8	-43	4.0	-50	11.6	-42	10.8	-40	2.9
7	11.1	-42	4.2	-50	11.1	-42	10.8	-40	2.6
8	8.7	-44	4.8	-49	10.8	-42	10.1	-41	2.3

Table (2.7). Average of absolute values of coefficients for different image categories. Transform size N= 16.

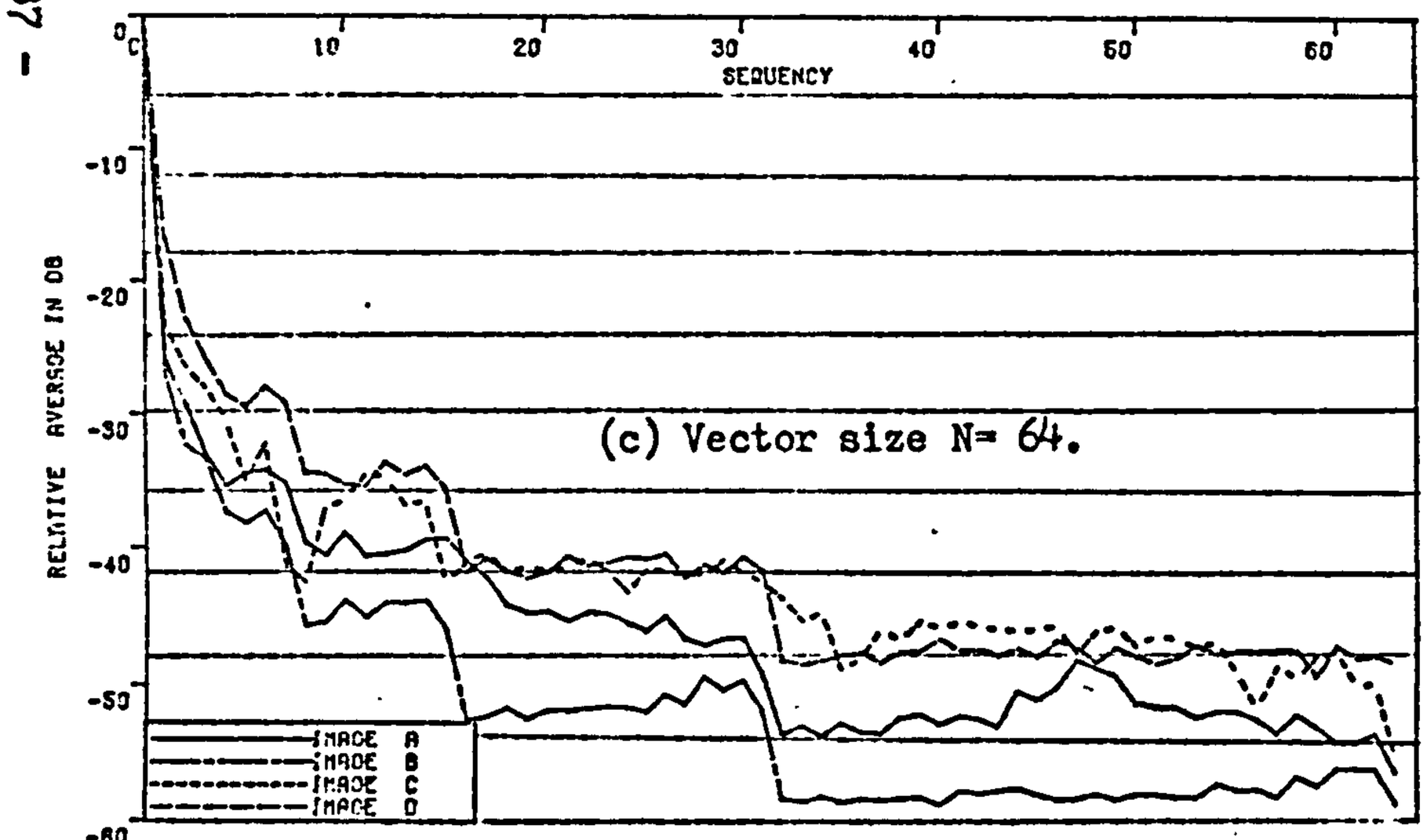
Image categ.	A		B		C		D		Ratio
Coeff. order.	Average.	Relative average, dB.	Average.	Relative average, dB.	Average.	Relative average, dB.	Average.	Relative average, dB.	Maximum to minimum.
1	2880	0	2620	0	2860	0	2210	0	1.3
2	102	-29	65	-32	115	-28	133	-24	2.0
3	62	-33	27	-40	70	-32	67	-30	2.6
4	61	-33	32	-38	60	-34	66	-31	2.1
5	38	-38	12	-47	35	-38	34	-36	3.2
6	32	-39	12	-47	41	-37	35	-36	3.4
7	31	-39	13	-46	37	-38	34	-36	2.8
8	27	-40	16	-44	33	-39	32	-37	2.1
9	13	-47	5.6	-53	17	-44	16	-43	3.0
10	13	-47	5.8	-53	17	-45	16	-43	2.9
11	14	-47	5.9	-53	19	-44	16	-43	3.2
12	17	-45	5.9	-53	17	-45	17	-43	2.9
13	17	-45	6.1	-53	18	-44	16	-43	3.0
14	15	-46	6.2	-53	20	-43	17	-43	3.2
15	14	-46	6.9	-52	17	-44	16	-43	2.5
16	12	-48	8.0	-50	14	-46	15	-43	1.9



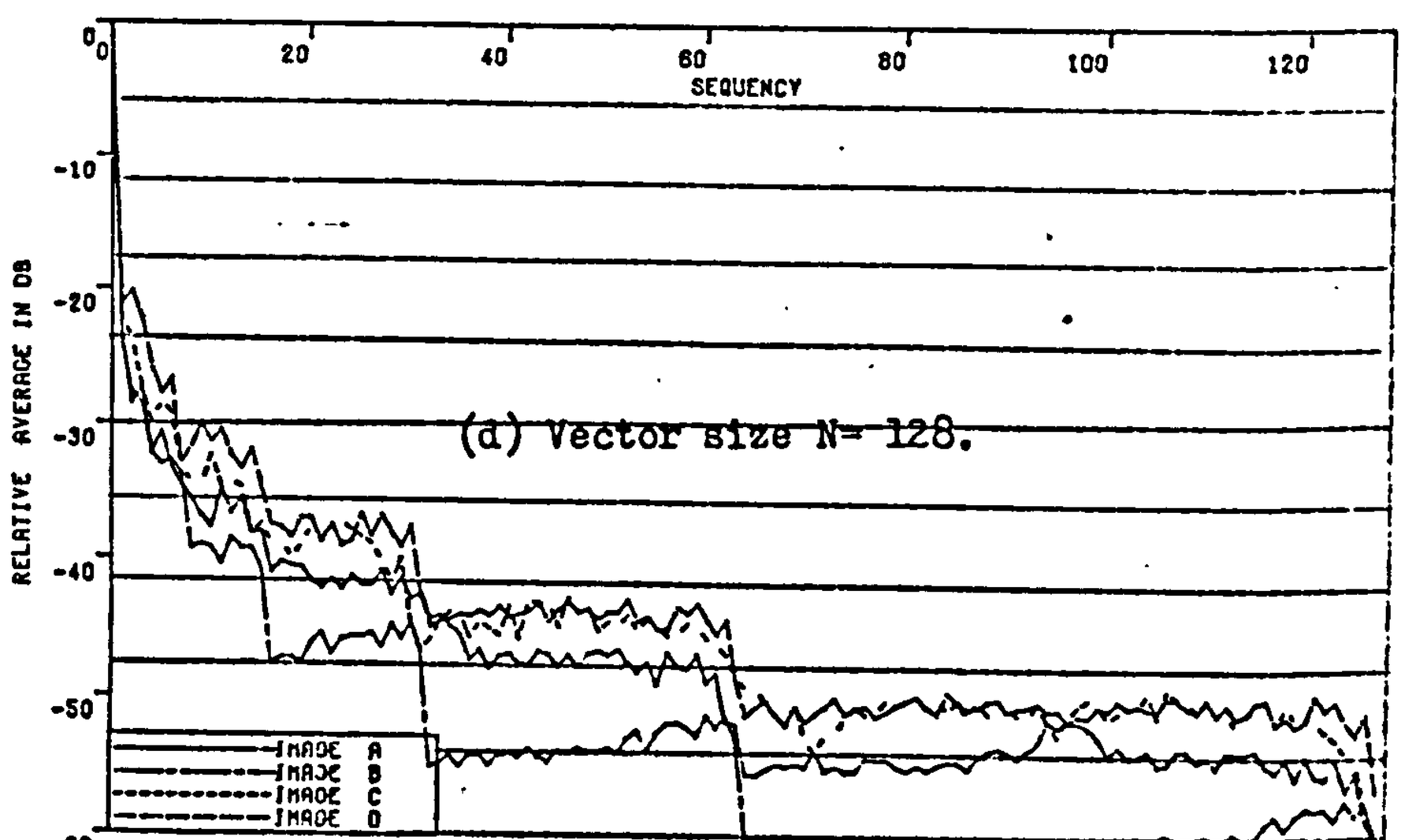
(a) Vector size N= 16.



(b) Vector size N= 32.



(c) Vector size N= 64.



(d) Vector size N= 128.

Figure (2.5). Averages of absolute values of coefficients in different transform sizes and difeerent images.

From these tables and figures, it is again clear that, generally speaking, values of coefficients decrease with the increase in their order (or sequency). Although this is true for all image categories, the values themselves are very strongly dependent on these categories. A measure of variation is included in the tables to show the ratio between maximum and minimum absolute values of a particular coefficient among the considered four image categories. Variations of up to 340% are noted in the tables.

It is to be noted from the tables and figures, that as the transform size increases, and consequently the number of coefficients, more and more ac coefficient values decrease sharply. Hence, less and less proportion of energy is associated with these coefficients. For example, for category C, while none of coefficients has a relative value less than-42 dB, in the case of a transform size of N=8, more than 75% of the coefficients have their values less than the same level (42 dB) in the case of a transform size of N=128. As an example of variations among different categories, the case of a transform size of N=32 and a threshold of-48 dB for coefficient relative values is taken now as an illustrative example. For category A, 44% of coefficients are lower than this threshold, for category B, about 70% , for category C only 3%, while no coefficient at all in category D is less than the same threshold. Tables (2.8 - 2.12) show a more comprhensive comparison for different categories and different transform sizes.

Table (2.8). Levels of transform coefficients values. N=8.

Levels of decrease than dc term.	Percent of coefficients with values within these levels.			
	A	B	C	D
18 - 24 dB	-	-	-	-
24 -30 dB	-	-	12.5	12.5
30 - 36dB	25	-	25	25
36 -42 dB	25	12.5	50	50
42 -48 dB	37.5	25	-	-
48 -54 dB	-	50	-	-
54 -60 dB	-	-	-	-

Table (2.9). Levels of transform coefficients values. N= 16.

Levels of decrease than dc term.	Percent of coefficients with values within these levels			
	A	B	C	D
18- 24 dB	-	-	-	6.25
24- 30 dB	6.25	-	6.25	6.25
30- 36 dB	12.5	6.25	12.5	25
36- 42 dB	25	12.5	25	6.25
42- 48 dB	50	25	50	50
48- 54 dB	-	50	-	-
54- 60 dB	-	-	-	-

Table (2.10). Levels of transform coefficients values. N= 32.

Levels of decrease than dc term.	Percent of coefficients with values within these levels.			
	A	B	C	D
18- 24 dB	-	-	-	3.1
24- 30 dB	3.1	3.1	6.2	6.2
30- 36 dB	12.5	6.3	12.5	12.5
36- 42 dB	18.7	9.4	28.1	25
42- 48 dB	18.7	9.4	46.9	50
48- 54 dB	43.9	28.1	3.1	-
54-,60 dB	-	40.6	-	-

Table (2.11). Levels of transform coefficients values. N= 64.

Levels of decrease than dc term.	Percent of coefficients with values within these levels.			
18- 24 dB	-	-	1.6	1.6
24- 30 dB	3.1	1.6	3.1	6.3
30- 36 dB	7.8	3.1	9.4	14
36- 42 dB	14	6.3	17.2	23.4
42- 48 dB	21.9	12.5	29.7	40.6
48- 54 dI	50	25	35.9	12.5
54- 60 dB	1.6	50	1.6	-

Table (2.12). Levels of transform coefficients values. N= 128.

Levels of decrease than dc term.	Percent of coefficients with values within these levels.			
	A	B	C	D
18- 24 dB	0.8	-	1.6	2.3
24- 30 dB	1.6	2.3	3.1	2.3
30- 36 dB	5.4	3.1	5.4	7.0
36- 42 dB	12.5	6.3	12.5	11.7
42- 48 dB	24.2	12.5	25.8	25
48- 54 dB	23.4	16.4	48.4	50
54- 60 dB	31.3	22.6	1.6	0.8
60- 66 dB	-	35.9	0.8	-

2.1.4. Energy Packing Efficiency: (64,65)

The whole idea of bit rate reduction using transformation is based on transmitting a limited number of coefficients in the transform domain rather than the whole set of samples in the spatial domain. These transmitted coefficients are selected because they have all or most of the luminance energy in the transformed area. Hence, a critical measure used in conjunction with the transform coding is "the Transform Energy Packing Efficiency".

In a study by Pratt(65), an "Energy Compaction Measure" was set as the percent of energy content in largest quarter of transform samples (coefficients), as compared to the total energy content in all coefficients. The value of 1/4 was rather an arbitrary value, and suggests what energy compaction efficiency is associated if a maximum bit rate reduction of a factor of 4 is attempted. Results presented there were for data modeled as a Markov process and not for actual image signals.

However, as a matter of suitability and practicality, it is felt here that a much more realistic packing efficiency measure would be one which deals with ac energy only. As the value of the dc term is always high, values of packing efficiencies including this term will be too close to each other. Hence, an accurate critical assessment of transform sizes or picture categories dependence will be prohibitively difficult. Therefore, in this study, an Energy Packing Efficiency for a limited number of coefficients will be defined as:

$$\frac{\text{A.C. energy contained in these coefficients}}{\text{Total ac energy contained in all coefficients}}$$

Instead of calculating the efficiency for an arbitrary proportion of coefficients, different values are computed for different proportions. For computing simplicity and suitability, although the energy associated with the dc term is not included, the coefficient itself is counted in calculating the proportion of coefficients.

Table (2.13) shows the energy packing efficiencies for a transform size of N= 16, and different picture categories. The first column shows the order of coefficient up to which the energy content is computed. First column in each image category shows the proportion of ac energy content, relative to the total ac energy in the whole transform area. For a quick comparison with the energy compacting measure mentioned above, the second column in each image category indicates the proportion of energy including that of dc term, compared with total transform energy.

Figures (2.6- 2.9) show ac energy packing efficiencies for different transform sizes, including that in Table (2.13), for different image categories. Figure (2.10) shows the ac energy packing efficiency for the "hypothetical image" composed of the four categories.

From these tables and figures, one can conclude that, generally, the ac energy packing efficiency is dependent on picture category. It is also clear that the efficiency increases with the transform size.An important fact is apparent, which again enhances the principles of "abruptancy" among certain groups of coefficients. Although the packing efficiency increases with coefficients sequency, the rate of increase is itself decreasing as we move from one group of coefficients to another higher order one. This well coincides with what have been noticed in Section(2.1.3)on absolute values of coefficients. As a result, an appreciable addition of energy content is normally achieved by including coefficients of a whole group. Moreover, the increase in energy content due to the highest half of coefficient order is very low especially for bigger transform sizes.

Table (2.13). Energy Packing Efficiencies for a transform size of N= 16, and different images.

Category.	A		B		C		D	
Up to coeff. no.	AC energy content %	All energy content %.	AC energy content %.	All energy content %.	AC energy content %.	All energy content%.	AC energy content %.	All energy content %.
1	0	99.02	0	99.66	0	98.43	0	98.49
2	31.4	99.46	40.9	99.86	32.8	99.02	38.8	99.31
3	47.2	99.61	53.0	99.90	44.9	99.35	53.9	99.51
4	61.7	99.76	66.2	99.95	56.1	99.57	67.1	99.71
5	70.1	99.83	70.7	99.96	61.6	99.64	73.4	99.76
6	76.2	99.87	75.0	99.97	67.3	99.72	78.9	99.81
7	81.4	99.90	79.3	99.97	72.6	99.80	83.7	99.86
8	85.3	99.93	84.0	99.99	78.1	99.86	87.7	99.91
9	87.3	99.94	86.1	99.99	81.7	99.89	89.4	99.92
10	89.3	99.95	88.1	99.99	84.6	99.90	91.1	99.93
11	91.2	99.95	90.3	99.99	87.4	99.92	92.8	99.94
12	93.4	99.97	92.4	99.99	90.1	99.94	94.5	99.96
13	95.5	99.98	94.4	99.99	93.0	99.96	96.2	99.97
14	97.2	99.99	96.3	100	95.6	99.97	97.7	99 98
15	98.9	99.99	98.2	100	97.9	99.99	99.0	99.99
16	100	100	100	100	100	100	100	100

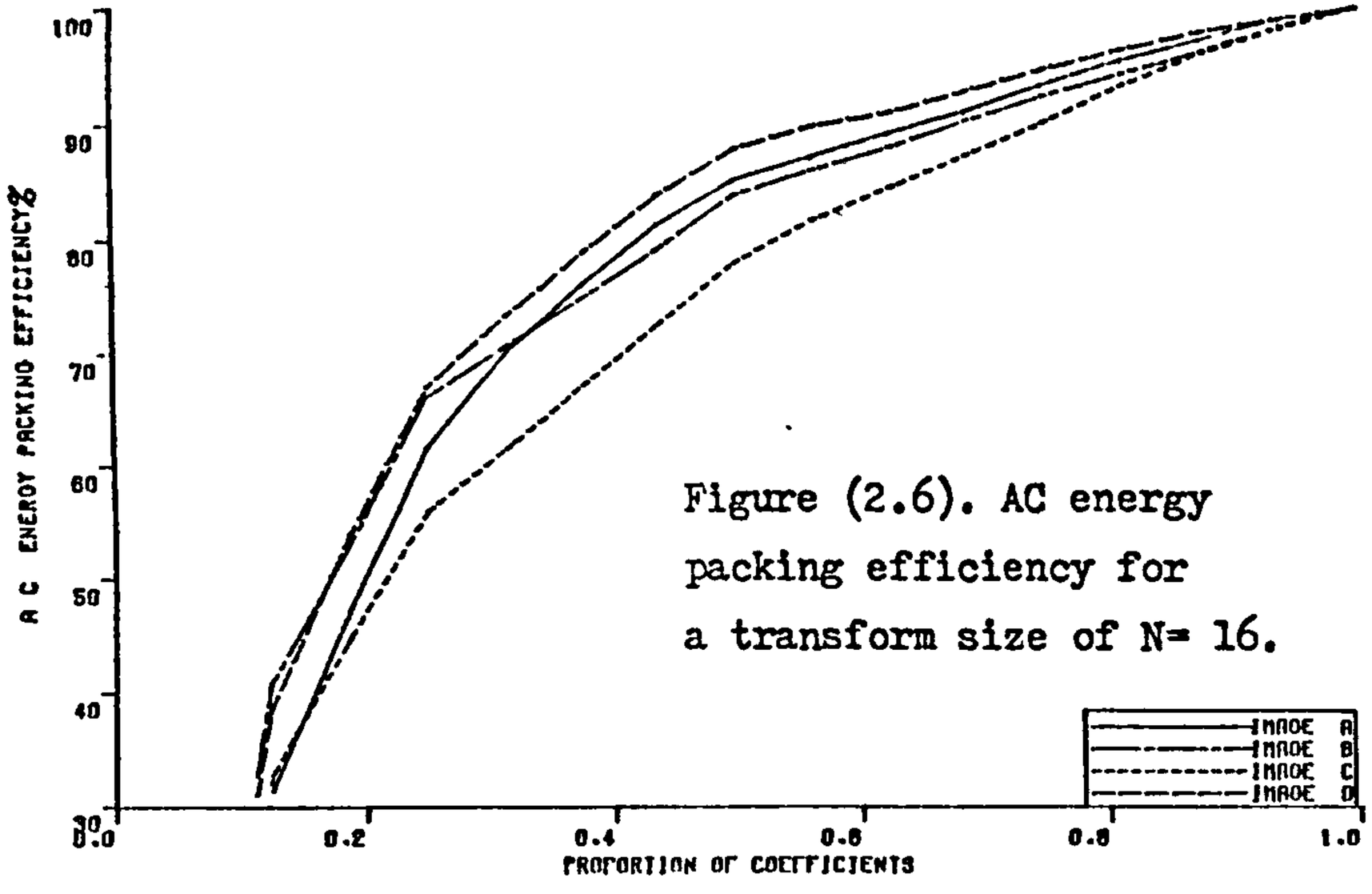


Figure (2.6). AC energy packing efficiency for a transform size of N= 16.

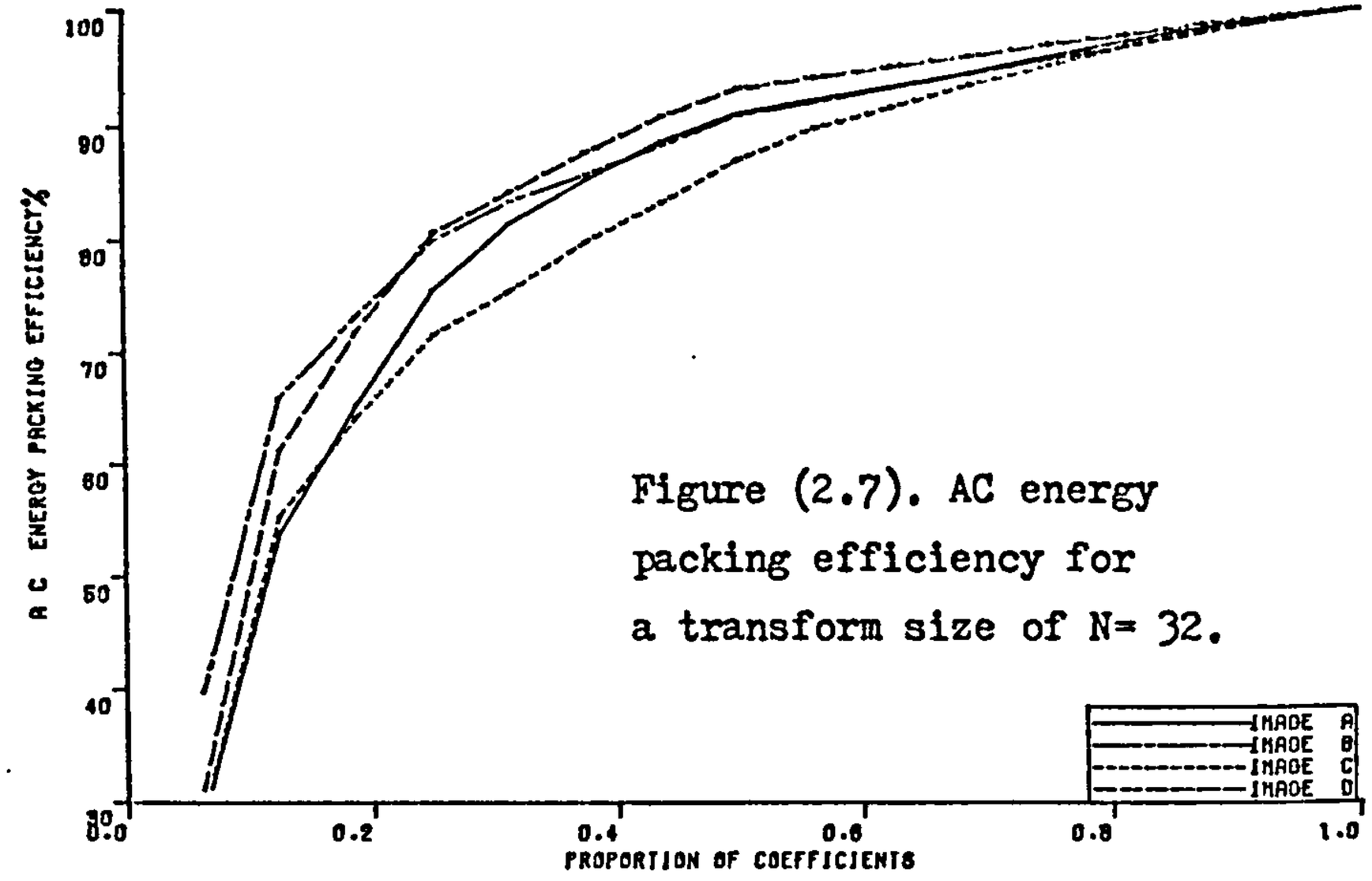


Figure (2.7). AC energy packing efficiency for a transform size of N= 32.

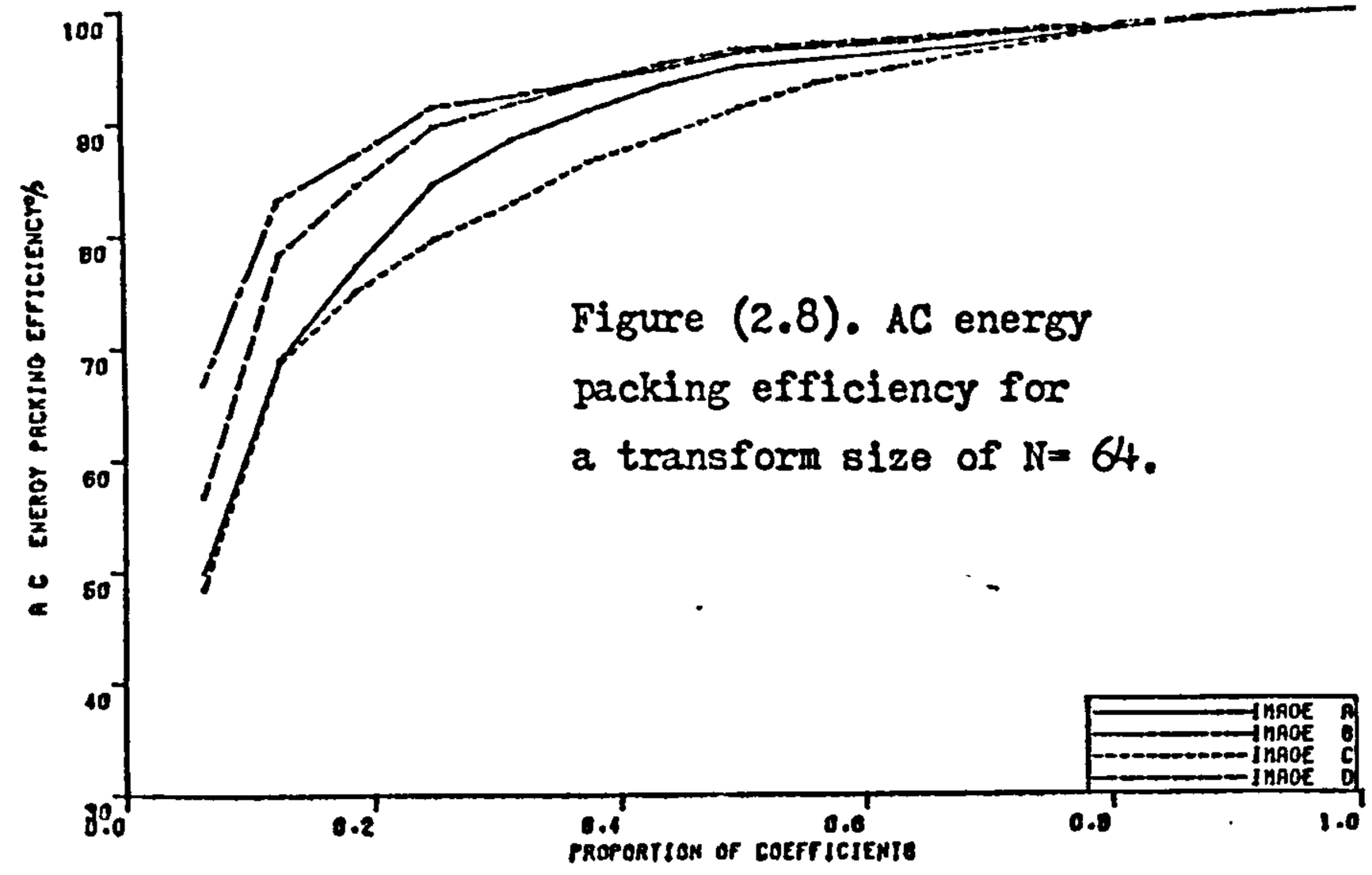


Figure (2.8). AC energy packing efficiency for a transform size of N= 64.

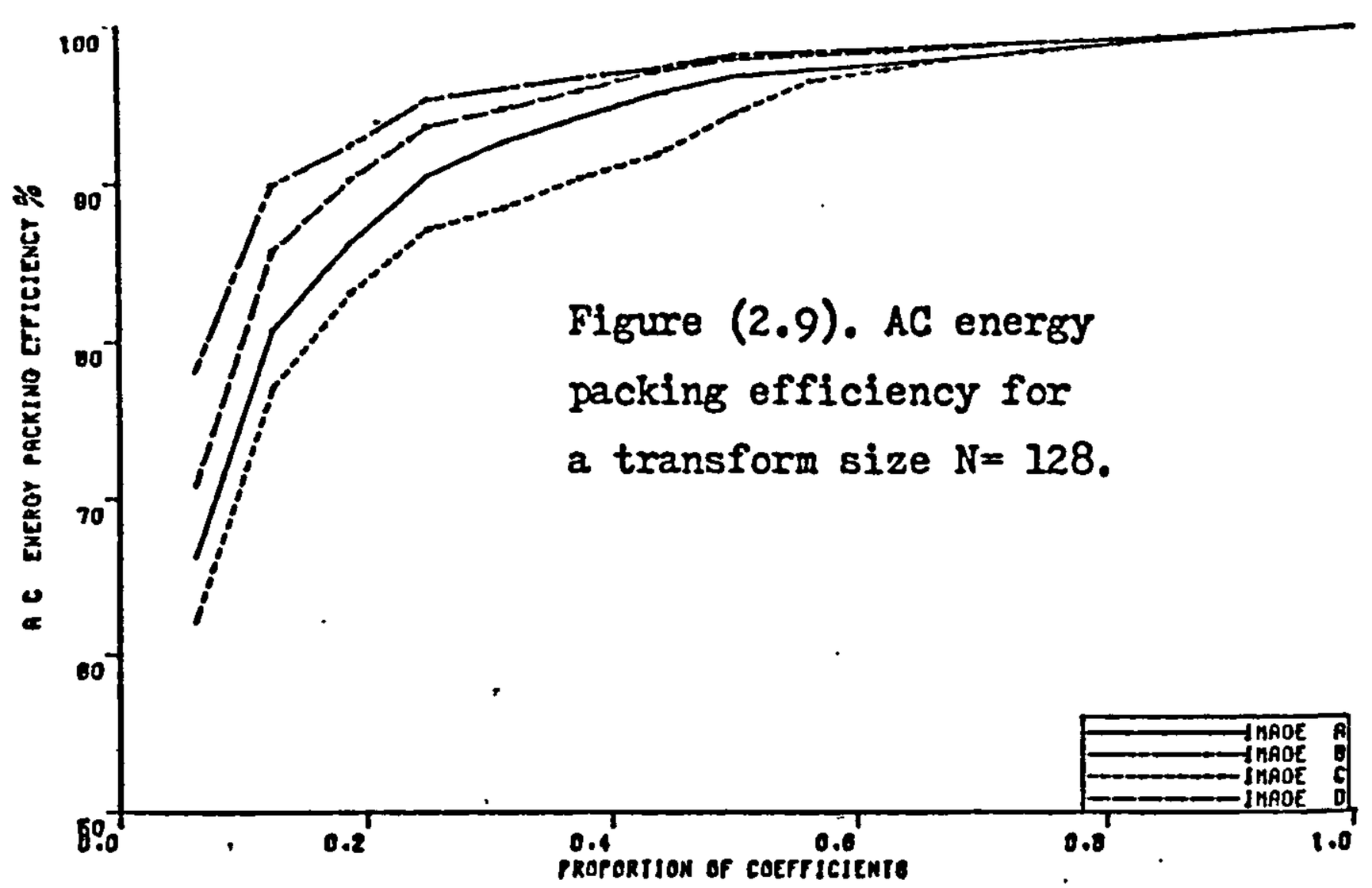


Figure (2.9). AC energy packing efficiency for a transform size N= 128.

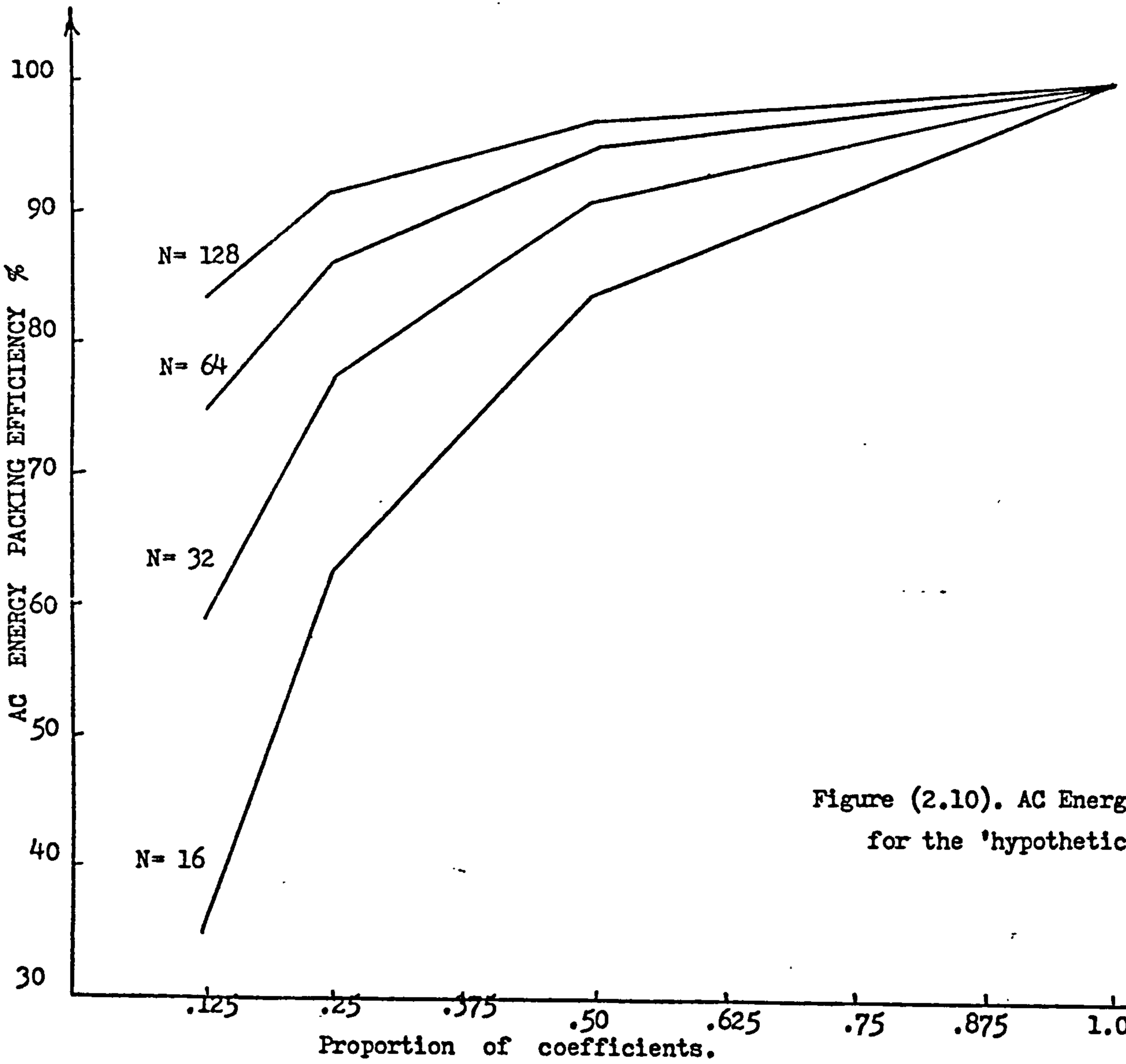


Figure (2.10). AC Energy packing Efficiency for the 'hypothetical' image.

2.1.5. Abruptancy in Transform Domain:

From results discussed earlier on absolute values and energy packing, it is clear that, in addition to the fact that transform coefficients decrease in values as their order (or sequency) increases the rate of decrease itself is not uniform alongside the transform spectrum. Some kind of "Abruptancy" exists in decreasing coefficients values and other statistics. Coefficients belonging to the same "Binary Bound Subvectors" seem to have, nearly in average, the same level of magnitudes. By "binary bound subvectors", we mean these subvectors resulting from setting binary boundaries to divide the transform vector. These boundaries are set at fractions $(1/2)^{n-m+1}$ of a vector of size $N=2^n$, where m=1, 2, 3, ... , n.

These n boundaries will give rise to (n+1) subvectors, each with different number and orders of coefficients (except for the first two). The first subvector will contain only the first (dc) coefficient H1, while the second will contain only the second coefficient H2 (of sequency one). Starting with the third subvector, the i^{th} subvector will have as its members the coefficients:

$H(2^{i-2}+1), H(2^{i-2}+2), \ldots\ldots\ldots\ldots\ldots, H(2^{i-1})$

where $i=3,4,\ldots\ldots,n$.

As an illustrative example, if n=4, then a vector of size $N=2^n$ (i.e. 16), will have (4+1)= 5 subvectors as shown in Figure (2.11).

Sequency:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Coeffic. order	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Figure (2.11). Binary boundaries and subvectors of a 16-pixel vector.

Abruptancy could be explained using the definitions of the transform and sequency mentioned in Section (1.3.). All transform coefficients, except the first one H_1, are composed of the sum of differences between couples of samples in spatial domain. Recalling the Hadamard matrix (Section 1.3.),there are always a constant number of additions and subtractions involved in each ac coefficient. This number is always equal to half the transform (vector) size. Therefore, and as seen from the matrix, each ac transform coefficient is in fact the result of adding half the samples values and subtracting the other half. It is only the positions of those samples added and those subtracted which determines the order (or sequency) of the coefficient and, generally, its range of values.

Recalling the same example as in Section(1.5.1) with N= 8, and writing values of transform coefficients H's as functions of samples values A's, the following set of equations will result:

$$H_2=A_1+A_2+A_3+A_4-A_5-A_6-A_7-A_8$$
$$H_3=A_1+A_2-A_3-A_4-A_5-A_6+A_7+A_8$$
$$H_4=A_1+A_2-A_3-A_4+A_5+A_6-A_7-A_8$$
$$H_5=A_1-A_2-A_3+A_4+A_5-A_6-A_7+A_8$$
$$H_6=A_1-A_2-A_3+A_4-A_5+A_6+A_7-A_8$$
$$H_7=A_1-A_2+A_3-A_4-A_5+A_6-A_7+A_8$$
$$H_8=A_1-A_2+A_3-A_4+A_5-A_6+A_7-A_8$$

Where H's = transform coefficients, and

A's = spatial domain samples values.

Rearranging these coefficients values as differences between samples, and grouping each subtracted sample with the nearest positive sample yields the results shown in Table (2.14).

Table (2.14). Values of ac coefficients as differences between nearest pairs of samples.

Coefficient	Value as a function of samples	Spaial distance
H_2	$(A_1-A_5) + (A_2-A_6) + (A_3-A_7) + (A_4-A_8)$	4 pixels
H_3	$(A_1-A_3) + (A_2-A_4) + (A_7-A_5) + (A_8-A_6)$	2 pixels
H_4	$(A_1-A_3) + (A_2-A_4) + (A_5-A_7) + (A_6-A_8)$	2 pixels
H_5	$(A_1-A_2) + (A_4-A_3) + (A_5-A_6) + (A_8-A_7)$	1 pixel
H_6	$(A_1-A_2) + (A_4-A_3) + (A_6-A_5) + (A_7-A_8)$	1 pixel
H_7	$(A_1-A_2) + (A_3-A_4) + (A_6-A_5) + (A_8-A_7)$	1 pixel
H_8	$(A_1-A_2) + (A_3-A_4) + (A_5-A_6) + (A_7-A_8)$	1 pixel

It is clear from the table above, that at low sequencies, the spatial distances between samples subtracted from each other are large, hence terms will be large differences resulting large values of low sequency coefficients. On the other hand, in case of higher sequencies, spatial distances are very small (direct or near direct neighbourhood), so that the differences between samples are close to zero, due to the inherent redundancy in TV signals. Therefore, values of high sequency coefficients are relatively low. Moreover, if redundancy increases, differences between neighbouring samples will decrease more and more, and hence the transform ac coefficients will have less values. It is to be noted also, from the table above, that the spatial distances are constant for all coefficients in the same binary bound subvector as indicated both in the left and right columns of the table. This explains the abruptancy in decreasing coefficient values from a subvector to another, while the average value within the same subvector is nearly constant.

As a practical example, recalling some values from Section (2.1.3) for a vector size of N= 16. The relative means of coefficients in each subvector (relative to the dc term), are shown for two major types of

pictures, namely the portrait and other remaining categories, in Table (2.15).

Table (2.15). Relative averages in binary bound subvectors.

	Average in a subvector, relative to H_1			
Subvector	2	3	4	5
Portrait	2.5×10^{-2}	1×10^{-2}	5×10^{-3}	2.5×10^{-3}
All others	4×10^{-2}	2×10^{-2}	1×10^{-2}	5×10^{-3}

For simplicity in dealing with dynamic ranges and coding of coefficients, again a logarithmic measure may be more expressive. Therefore, an "Abruptancy Measure" is set to judge the trend of decreasing coefficients values with the order of subvectors. This measure is defined as : " the drop, expressed in dB, in average relative values of coefficients within the subvector concerned, compared with the average relative values of coefficients within the immediately preceding subvector ".

In symbolic notation:

$$A_i = 20 \log_{10} C_i / C_{i-1} \quad \text{dB}$$

where: A_i = abruptancy measure of subvector i

C_i = statistical mean of relative averages within subvector i

C_{i-1} = statistical mean of relative averages within subvector i-1

Applying this measure to results in Table (2.15), another table, Table (2.16), shows the different values of abruptancies.

Table (2.16). Abruptancy among binary bound subvectors. N= 16.

Abruptancy. / Category.	A_2	A_3	A_4	A_5
Portrait.	32	6	6	6
Others.	24-28	6	6	6

From Table (2.16) above, it is clear that although A_2 varies considerably with image category, other abruptancies are nearly constant at 6 dB for all pictures. As far as the dynamic ranges and coding of coefficients are concerned, the high value of A_2 suggests that the decrease in number of bits allocated to coding H_2will be substantial, i.e.much less bits than dc coefficient. However, for higher order sub-vectors, the nearly constant value of abruptancies at 6 dB, suggests that a reduction of only one bit in the number of bits allocated to corresponding coefficients than coefficients in preceding subvectors is expected. Moreover, coefficients to be coded from one single sub-vector should be, in general, treated in similar way concerning the number of bits, and, may be, the coding schemes.

Another measure is set for the "Uniformity" in values of averages within a particular subvector. This measure is taken as the difference between minimum and maximum values of averages within the subvector, related to the overall average value in that subvector. As an example, for the same case considered earlier, the uniformities of subvectors 3, 4, and 5 for the different picture categories are as shown in Table (2.17).

Table (2.17). Uniformity within binary bound subvectors.

Sub-vector number:		Uniformity measures		
		3	4	5
Image :	A	2 %	34%	35%
	B	17%	30%	38%
	C	15%	22%	34%
	D	1.5%	9 %	12.5%

2.2. Block Transformation

Block transformation is the two-dimensional transform. This can be considered as the general case, and therefore it was the case which has been explained earlier in defining the transform in Section (1.3.) A limited number of TV lines are considered. This number represents the vertical dimension of a block. The length of the line is divided into a number of segments, each containing a limited number of samples. This number represents the horizontal dimension of the block.

Figure (2.12) shows the top 8 lines of a television picture. For a square block of N samples, number of samples along each side of the block will be $\sqrt{N}$. Therefore, $\sqrt{N}$ lines will be taken at first, grouped horizontally into $640/\sqrt{N}$ blocks. The figure shows a case where N=16, i.e. a block of 4x4 samples.

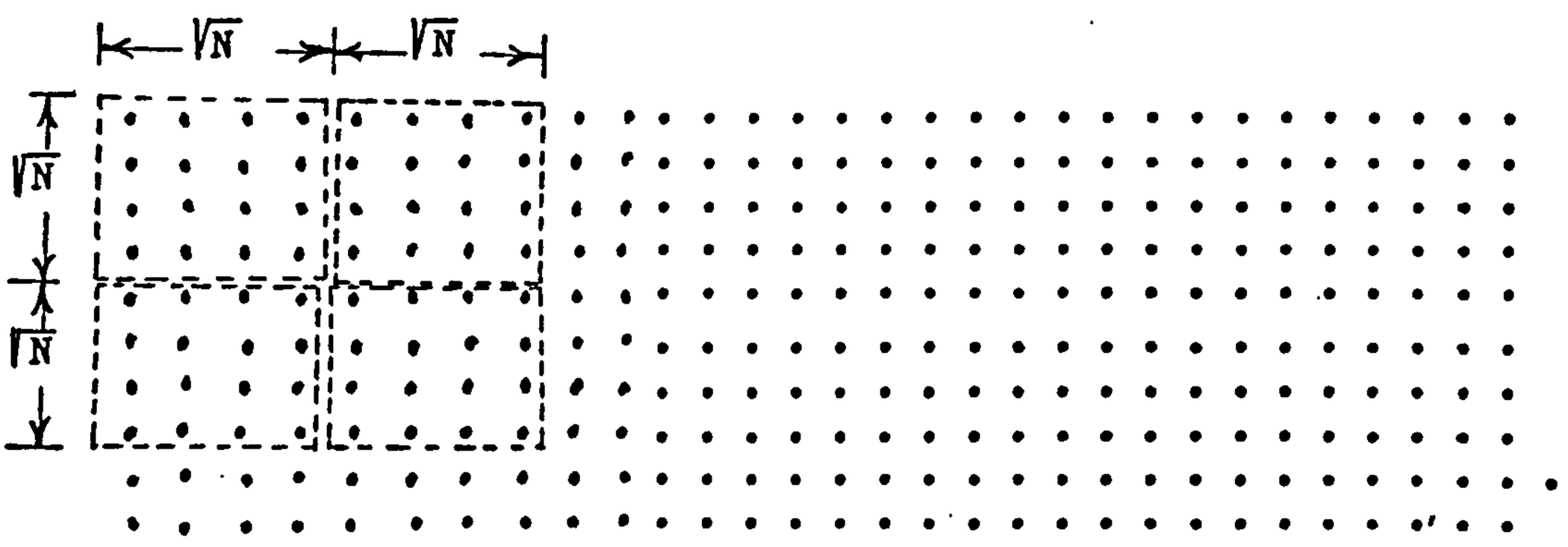
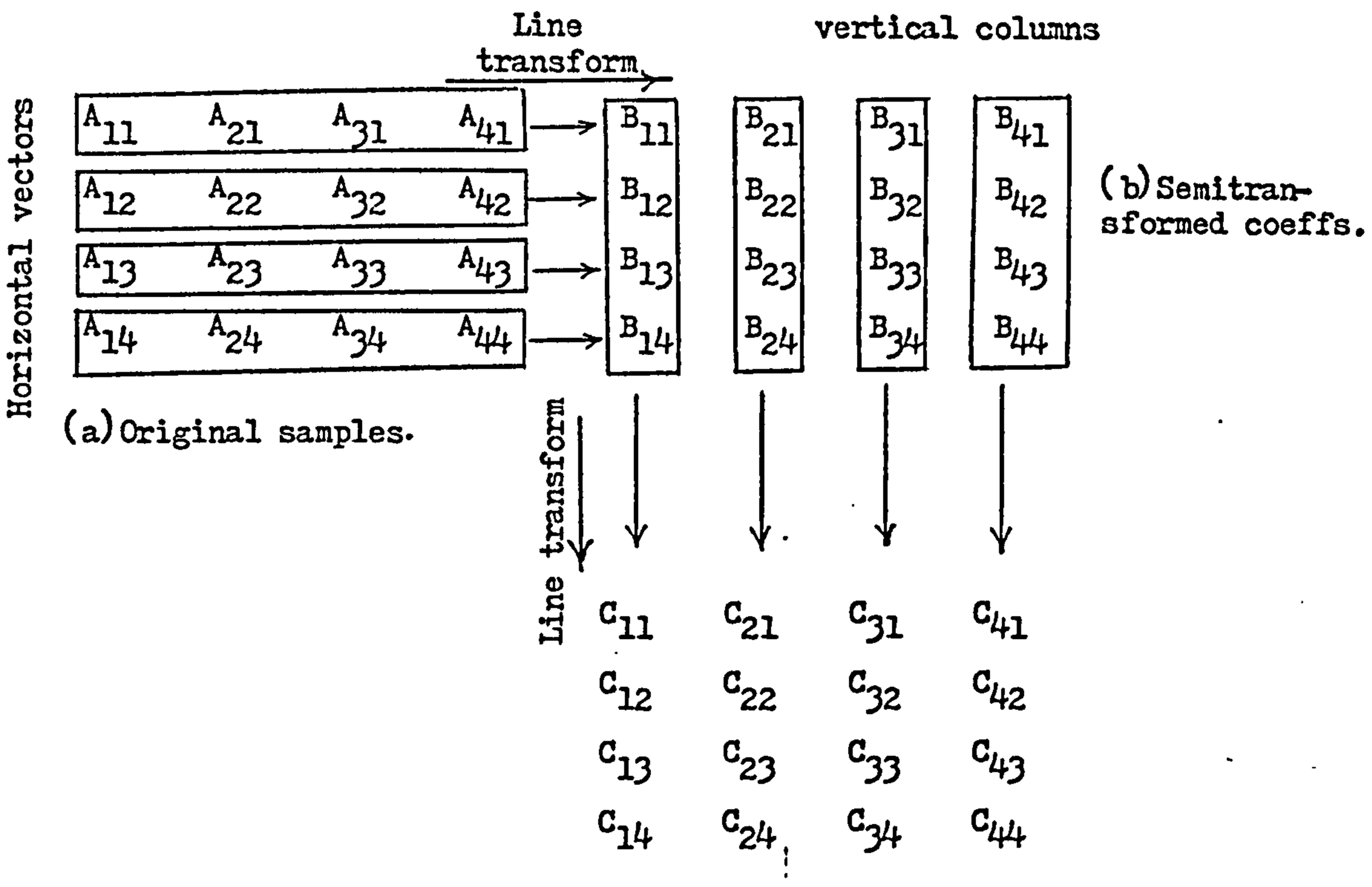


Figure (2.12). Grouping spatial samples in case of block transformation. Block size is 4x4.

As mentioned in Section (1.3.), the block transformation is the same as a double-transformation. First the 'horizontal vectors' of samples are line transformed as in Section(2.1), resulting 'semi-transformed' coefficients. Then the 'vertical columns' of semitransformed coefficients are again line transformed. The final result are the 'block transform coefficients'.



(a) Original samples.

(b) Semitransformed coeffs.

(c) Fully transformed coefficients.

Figure (2.13). Block transformation, as a double line transformation.

Figure (2.13) illustrates the sequence of two line transformations to yield the block transform. In (a), the spatial samples are represented by A_{ij}'s, and are grouped in four horizontal vectors along the i variable direction. Each of these vectors is line transformed, the results are B_{ij}'s which are shown in (b). These B_{ij}'s are again grouped in four vertical columns along the j variable direction. After line transforming these vertical columns, the final results are shown in (c) as coefficients C_{ij}'s, among which C_{11} is the only dc term for the whole block.

2.2.1. Probability Density Functions of Coefficients :

No apparent differences in PDF of block transform coefficients than in the case of line transform were noticed. PDF's are nearly similar for coefficients of corresponding orders in two equivalent size transforms.

2.2.2. Variances of Coefficients :

As was the case in line transformation, values of variances, and hence standard deviations, among different images categories vary considerably for the same block sizes.

Table (2.18) shows values for coefficient variances for a block size of 4x4 in different image categories. The table is directly compatible with that of line transform in the corresponding meanings of entries. However, in the block transformation case, the order of coefficients is shown in the first column with two subscripts, I for the horizontal direction, and J for the vertical direction.

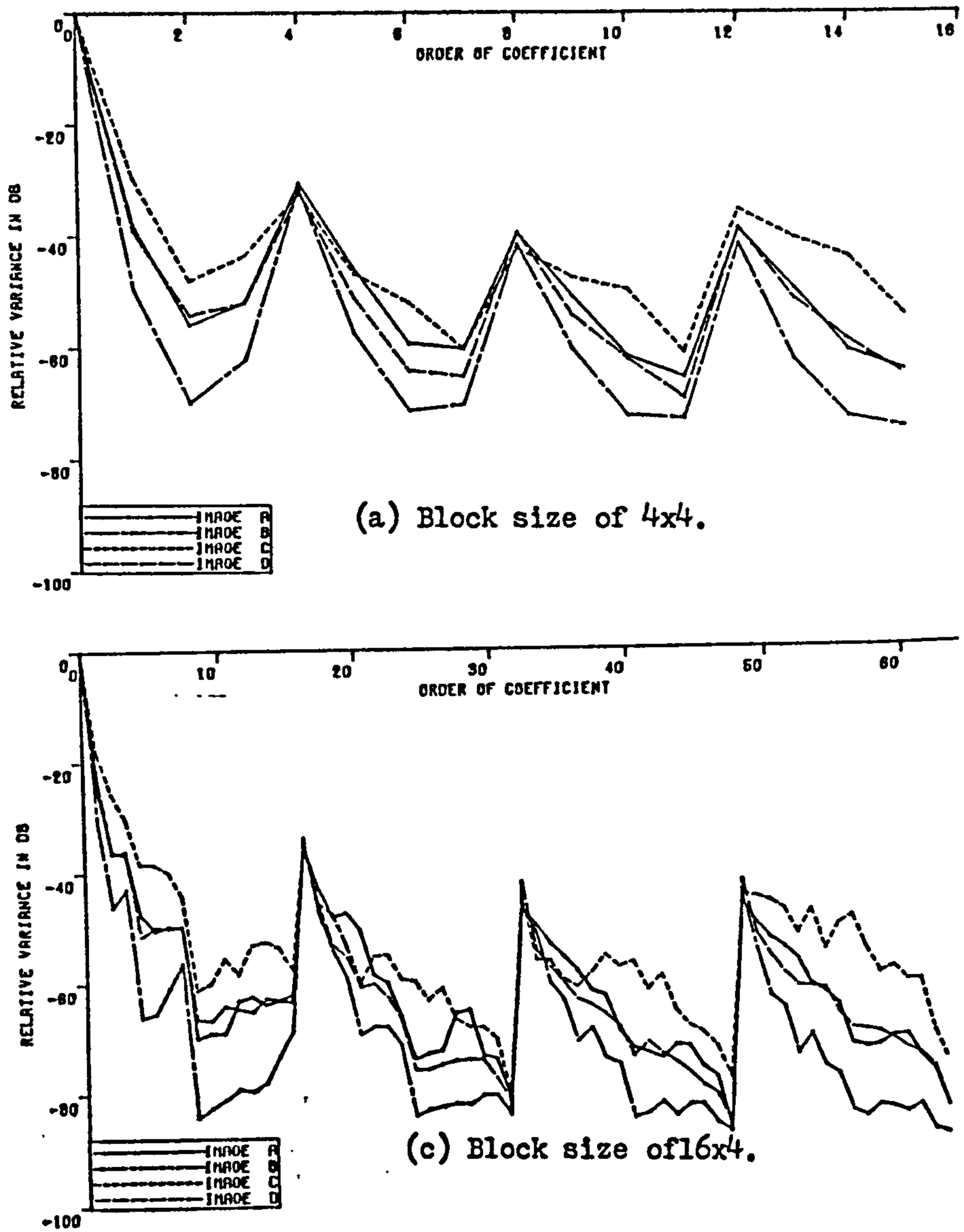
Direct comparison with line transform of equivalent size,(i.e.a transform which has the same number of coefficients), shows that in general, the differences in values of variances among different pictures are slightly lower in case of block transformation. Also the values of variances, and hence the standard deviations are much lower than in the line transform, except for the dc term , where the values are slightly larger.

Figure (2.14) shows relative variances for block transforms of different sizes. Orders of coefficients are related to coordinates i,j as : order= (j-1)Nx +i -1, where Nx is the horizontal dimension of the block. Periodicity noticed in the graphs is only due to the act of grouping together all the sequencies belonging to the same vertical coordinate (j).

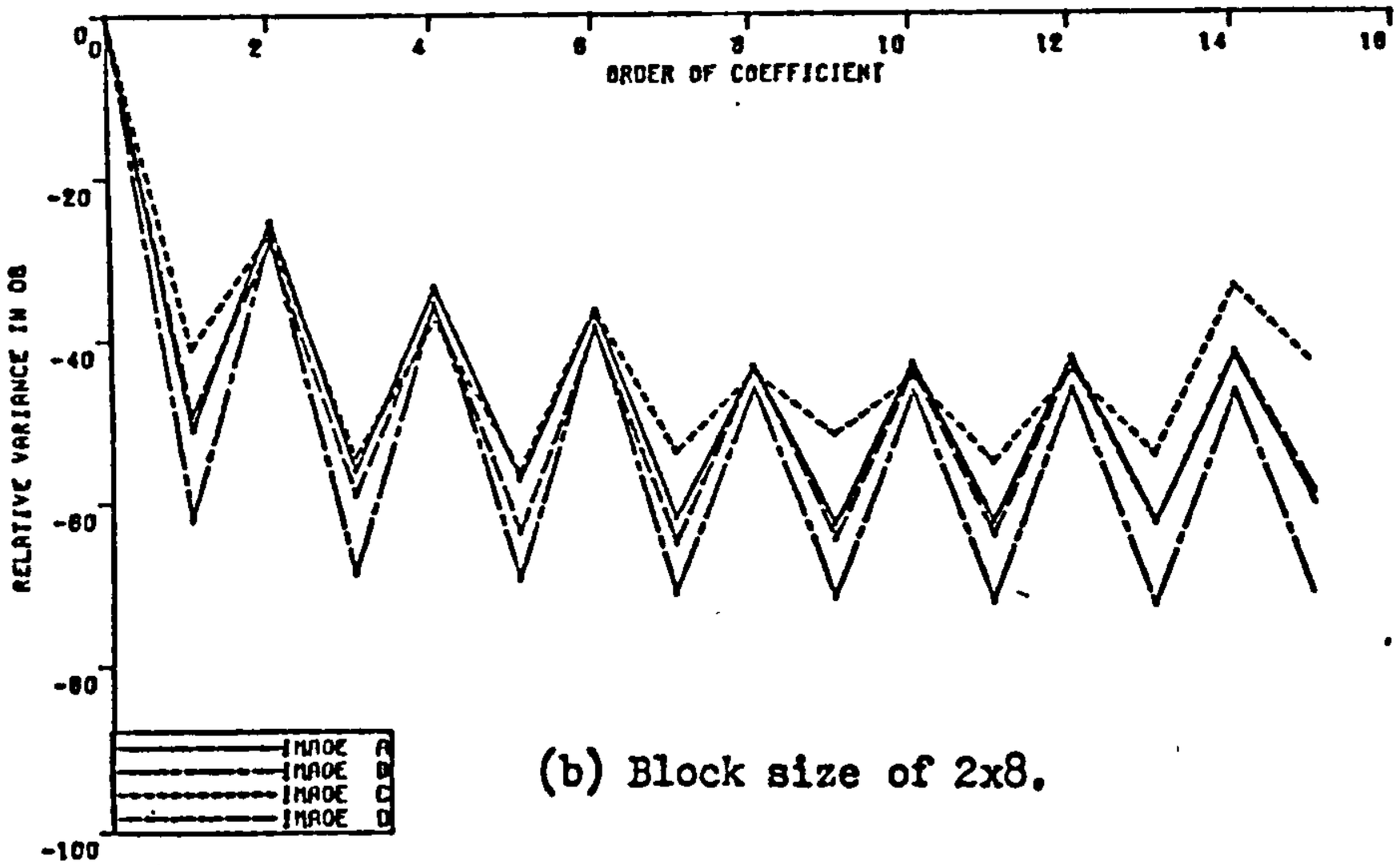
Table (2.18). Variances of different coefficients for different image categories. Block size = 4x4.

Category		A		B		C		D		Ratio
I	J	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	Variance.	Relative var. in dB.	Maximum/ Minimum.
1	1	452000	0	293000	0	339000	0	571000	0	1.95
2	1	5710	-38	987	-49	11500	-29	6360	-39	11.65
3	1	722	-56	96	-70	1340	-48	1120	-54	13.96
4	1	1150	-52	233	-62	2260	-44	1440	-52	9.70
1	2	13500	-30	7910	-31	8080	-33	13900	-32	1.76
2	2	2330	-46	395	-57	1560	-47	1590	-51	5.90
3	2	494	-59	80	-71	856	-52	358	-64	10.70
4	2	437	-60	92	-70	330	-60	314	-65	4.75
1	3	4800	-39	2480	-41	2600	-42	6120	-39	2.47
2	3	1300	-51	290	-60	1430	-47	1110	-54	4.93
3	3	382	-61	73	-72	1110	-50	454	-62	15.21
4	3	246	-65	68	-73	310	-61	205	-69	4.56
1	4	5200	-39	2480	-41	5860	-35	6760	-39	2.73
2	4	1540	-49	232	-62	3270	-40	1560	-51	14.10
3	4	433	-60	73	-72	2220	-44	671	-59	30.41
4	4	289	-64	58	-74	667	-54	333	-65	11.50

Figure (2.14). Relative variances of coefficients in different block sizes and different images.

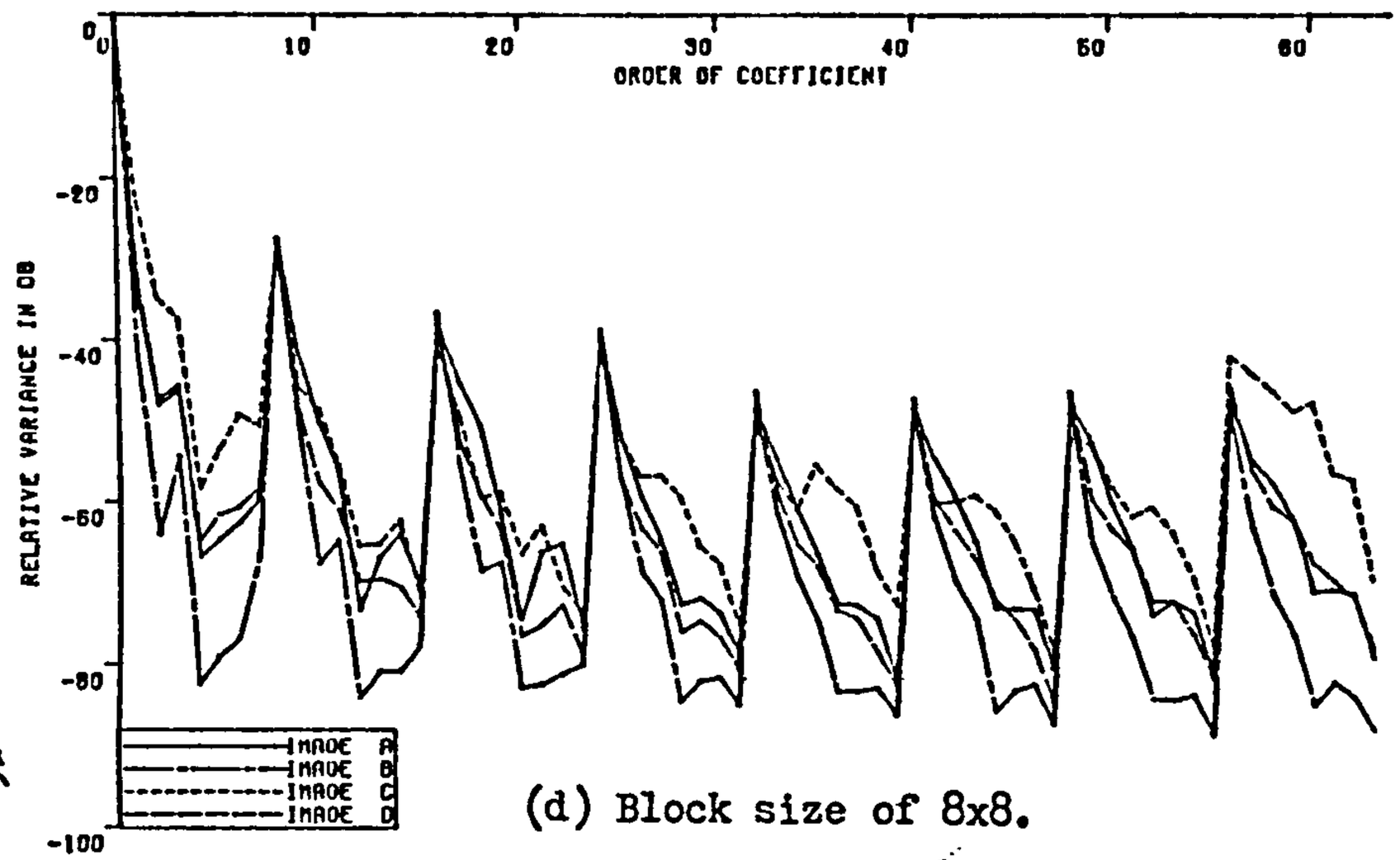


(a) Block size of 4x4.

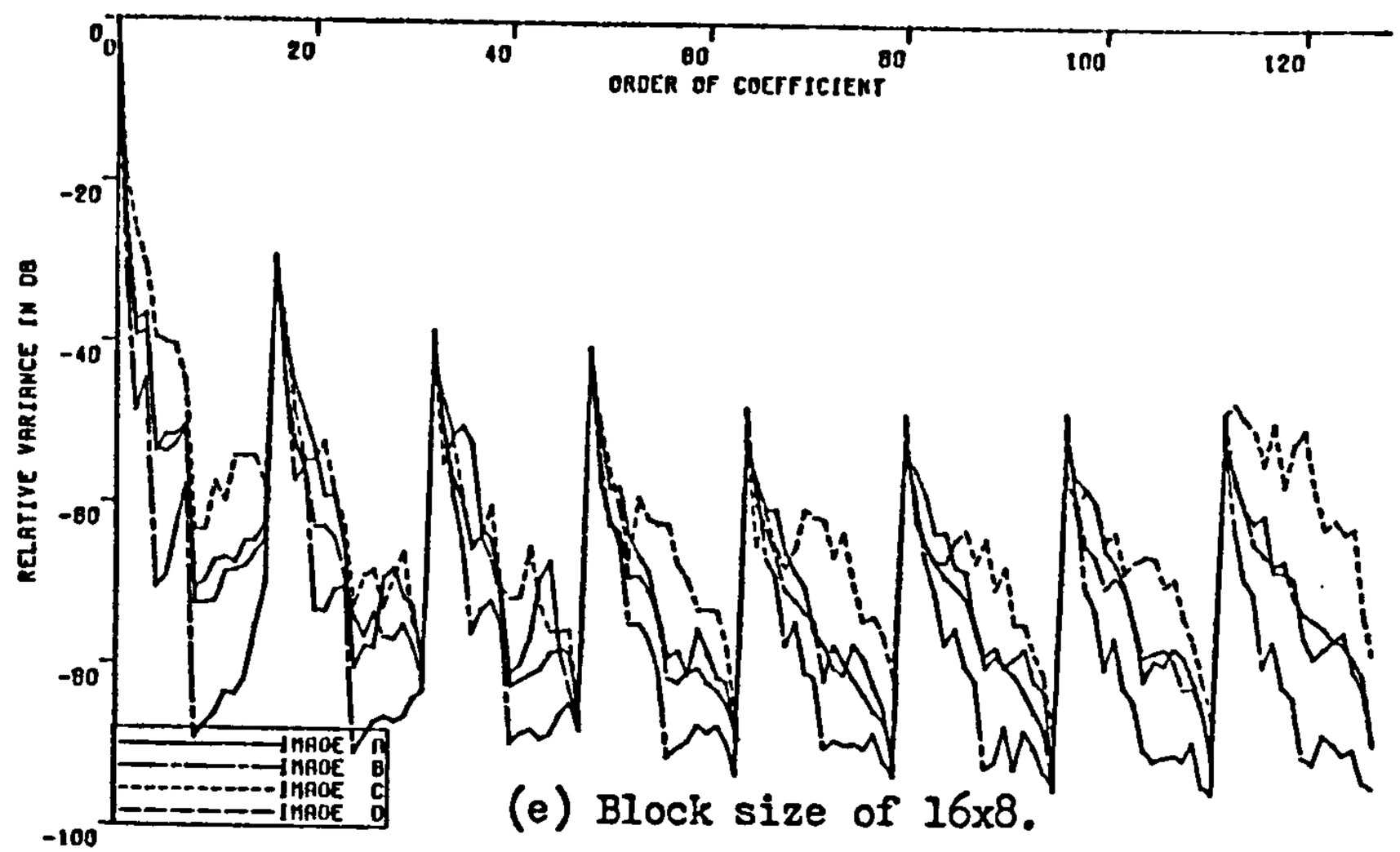


(b) Block size of 2x8.

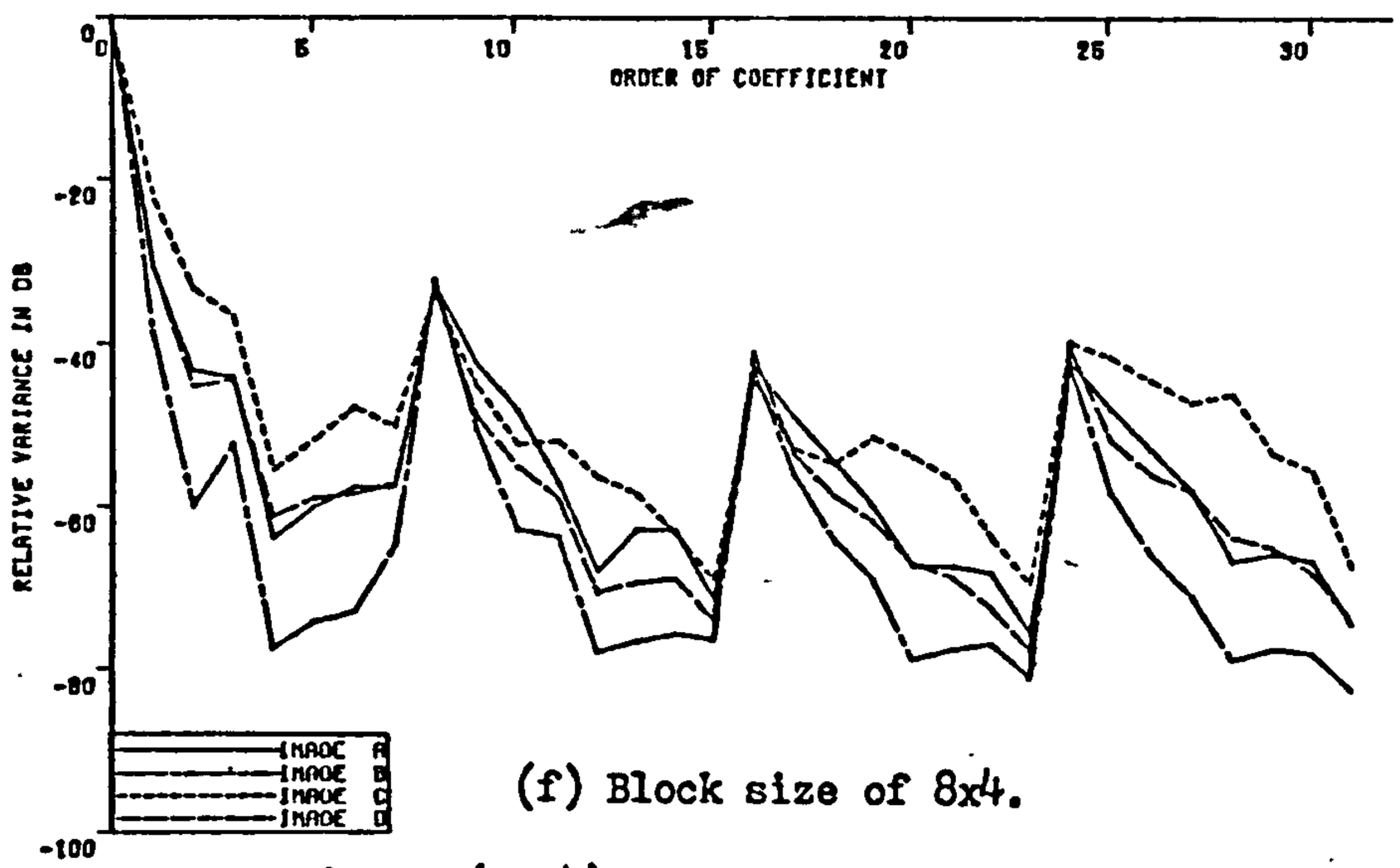
(c) Block size of16x4.



(d) Block size of 8x8.

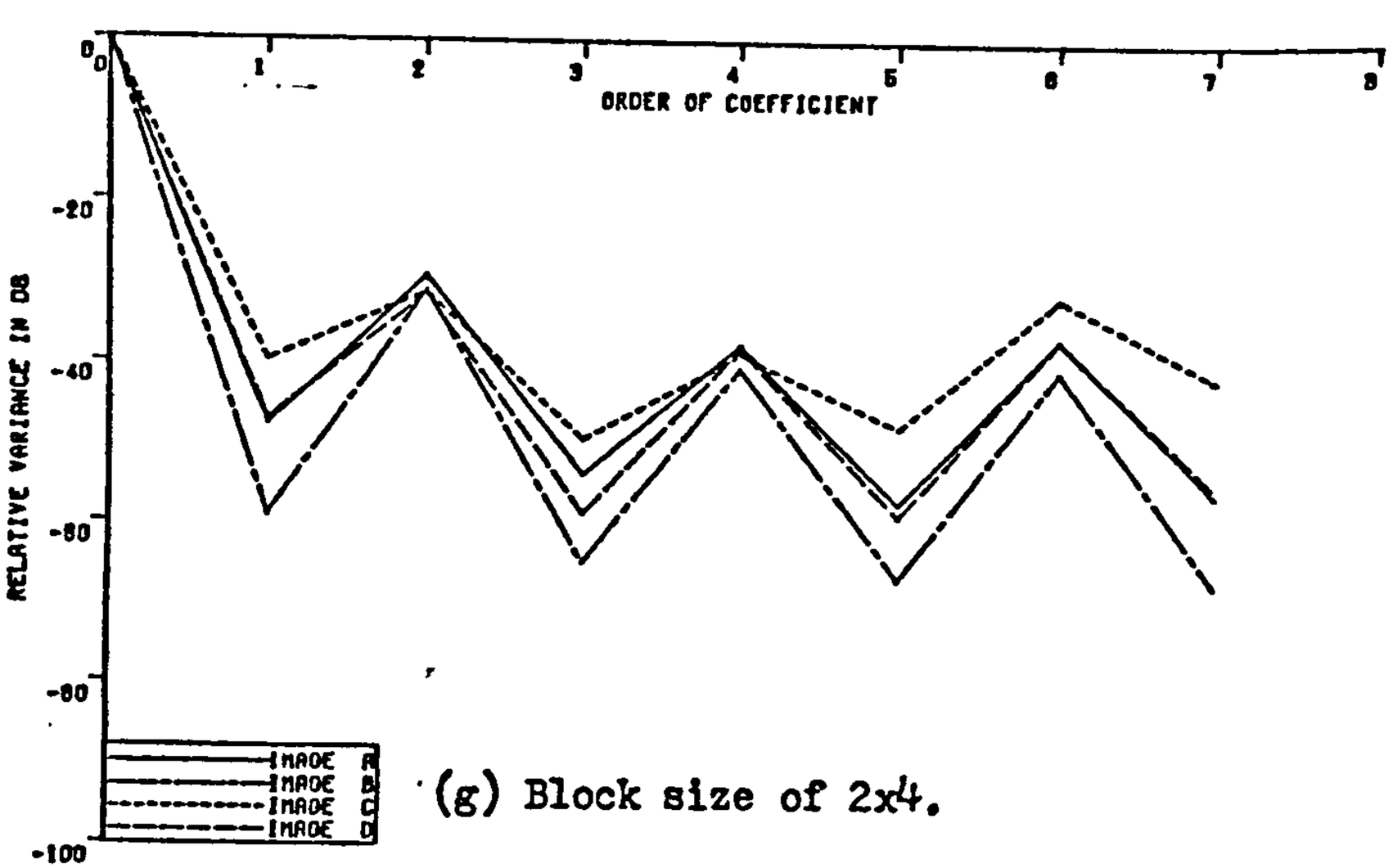


(e) Block size of 16x8.



(f) Block size of 8x4.

Figure (2.14) continued.



(g) Block size of 2x4.

2.2.3. Absolute Values of Transform Coefficients:

In dealing with two-dimensional transform, one should first have an idea about the manner the values of coefficients change at.In line transform, it has been shown that, in general, coefficients decrease in values as their sequencies increase. In block transform the increase in sequency is a monotical one, starting at the left hand upper corner of a block and ending at the right hand lower corner. Thinking of the block transform as two processes of line transformations, an approximate idea of the likely ranges of values of coefficients could be reached as follow.

Recalling results of Section (2.1.5), it can be assumed that average values of coefficients in a subvector are nearly half of that in a preceding subvector, except for subvector 2. Therefore, if the dc term value is assumed to be D, and the first ac coefficient (of sequency 1, or order 2)to be aD, (where a is a factor much less than unity), then the likely values of averages in subsequent subvectors could be estimated.

Figure (2.15) shows a line transform of size 8, as an example, and its subvectors divisions, and likely ranges of values of coefficients.

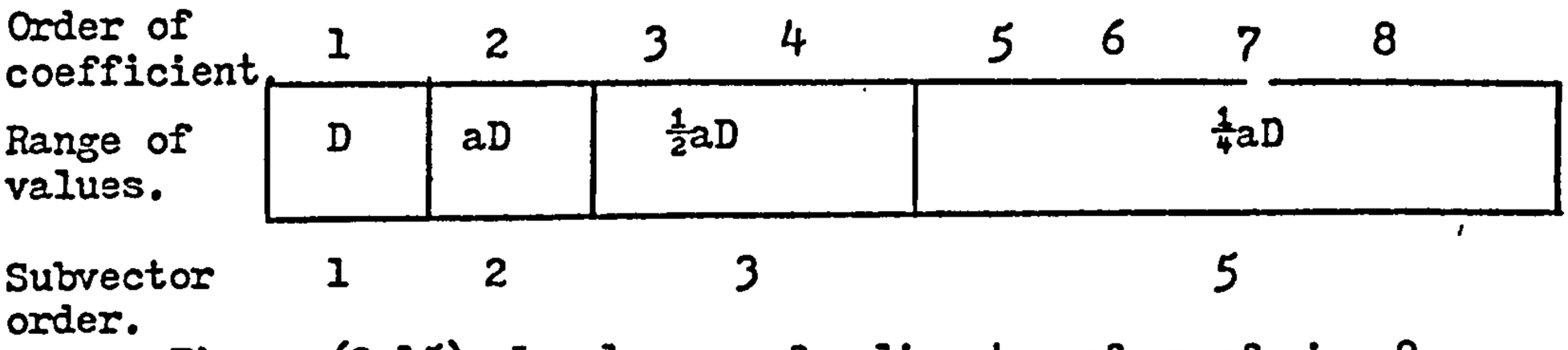


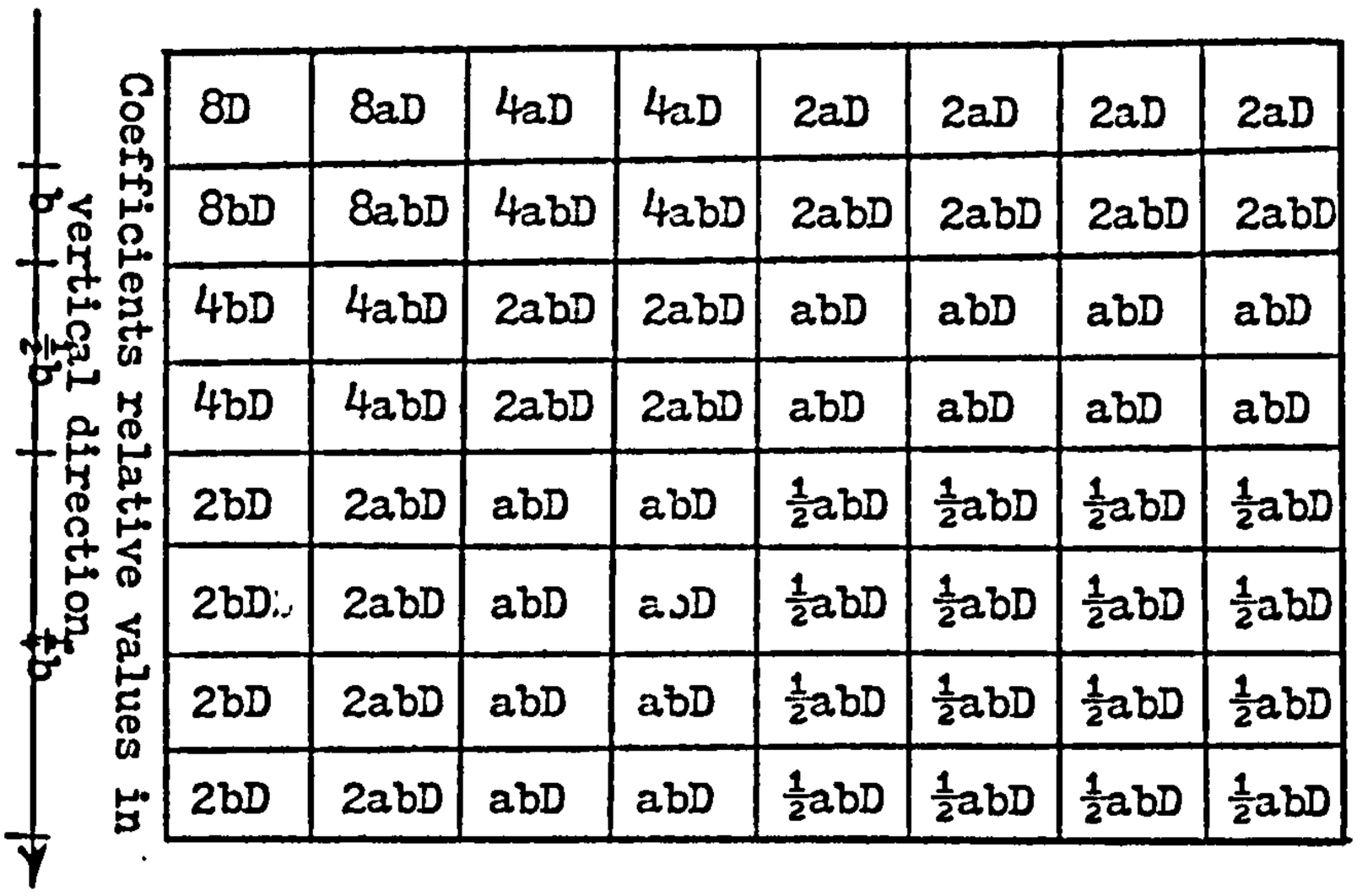
Figure (2.15). Levels map of a line transform of size 8.

Now taking an example of the block transform of 8x8, Figure (2.16) shows the likely values after first line transformation along horizontal direction. Applying the second line transform along the vertical direction on the semitransformed block will result in the finally transformed block of coefficients, with likely ranges of values shown in Figure (2.17), noting from definitions of the transform that the dc term is the sum of all

Coefficients relative values in horizontal direction.

D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD
D	aD	½aD	½aD	¼aD	¼aD	¼aD	¼aD

Figure (2.16). Levels map of ranges of values after horizontal transform.



8D	8aD	4aD	4aD	2aD	2aD	2aD	2aD
8bD	8abD	4abD	4abD	2abD	2abD	2abD	2abD
4bD	4abD	2abD	2abD	abD	abD	abD	abD
4bD	4abD	2abD	2abD	abD	abD	abD	abD
2bD	2abD	abD	abD	½abD	½abD	½abD	½abD
2bD	2abD	abD	abD	½abD	½abD	½abD	½abD
2bD	2abD	abD	abD	½abD	½abD	½abD	½abD
2bD	2abD	abD	abD	½abD	½abD	½abD	½abD

Figure (2.17). Levels map of ranges of values after vertical transform.

samples. Then the value of coefficient of order (1,1) will be in the range of 8D. Again assuming another factor b to approximate the value of next coefficient and a factor of ½ for each subsequent subvector in the vertical direction transformation, the likely levels map will be as shown in Figure (2.17).

From Figure (2.17), and assuming that a= b (which is a practical assumption), there are some areas in the block where the coefficients are likely to have the same ranges of values. Grouping these areas in subblocks yields the broken subblocks shown in Figure (2.18). In this figure, the map shows subblocks separated from each other and situated in their relative positions as in the original transform block. The orders of the subblocks, as far as values of coefficients are concerned, are shown within the subblocks. Number of coefficients and their subscripts can be known from the axes I,J given alongside the horizontal and vertical dimensions of the block, respectively.

Table (2.19) shows the subblocks, the likely ranges of values, and the coefficients subscripts within each subblock. Practical "levels maps" for the pictures analysed here are shown in Figures (2.19 &2.20). From these maps it seems that the drop in coefficients values in horizontal direction is generally more than the drop in vertical direction. This may be caused by different correlation factors in original data. It is also clear that although the theoretical ordering of coefficients is generally applicable, there are more variations among coefficients in particular

Table (2.19). Subblocks of Figure(2.18).

Subblock order	Range of values	No.of coeffs.	Subscripts (i,j)
1	8D	1	1,1
2	8aD	2	1,2&2,1
3	4aD,4bD	4	$(1,j)_{j=3,4}$ &$(i,1)_{i=3,4}$
4	2aD,2bD	8	$(1,j)_{j=5-8}$ & $(i,1)_{i=5-8}$
5	8abD	1	2,2
6	4abD	4	$(i,2)_{i=3,4}$ & $(2,j)_{j=3,4}$
7	2abD	12	$(i,2)_{i=5-8}$&$(2,j)_{j=5-8}$&$(i,j)_{i=3,4\ \&\ j=3,4}$
8	abD	16	$(i,j)_{i=5-8\ \&\ j=3,4}$ &$(i,j)_{i=3,4\ \&\ j=5-8}$
9	½abD	16	$(i,j)_{i=5-8\ \&\ j=5-8}$

subblocks. This suggests that, unlike the line transform, there is no unique order of block transform coefficients.

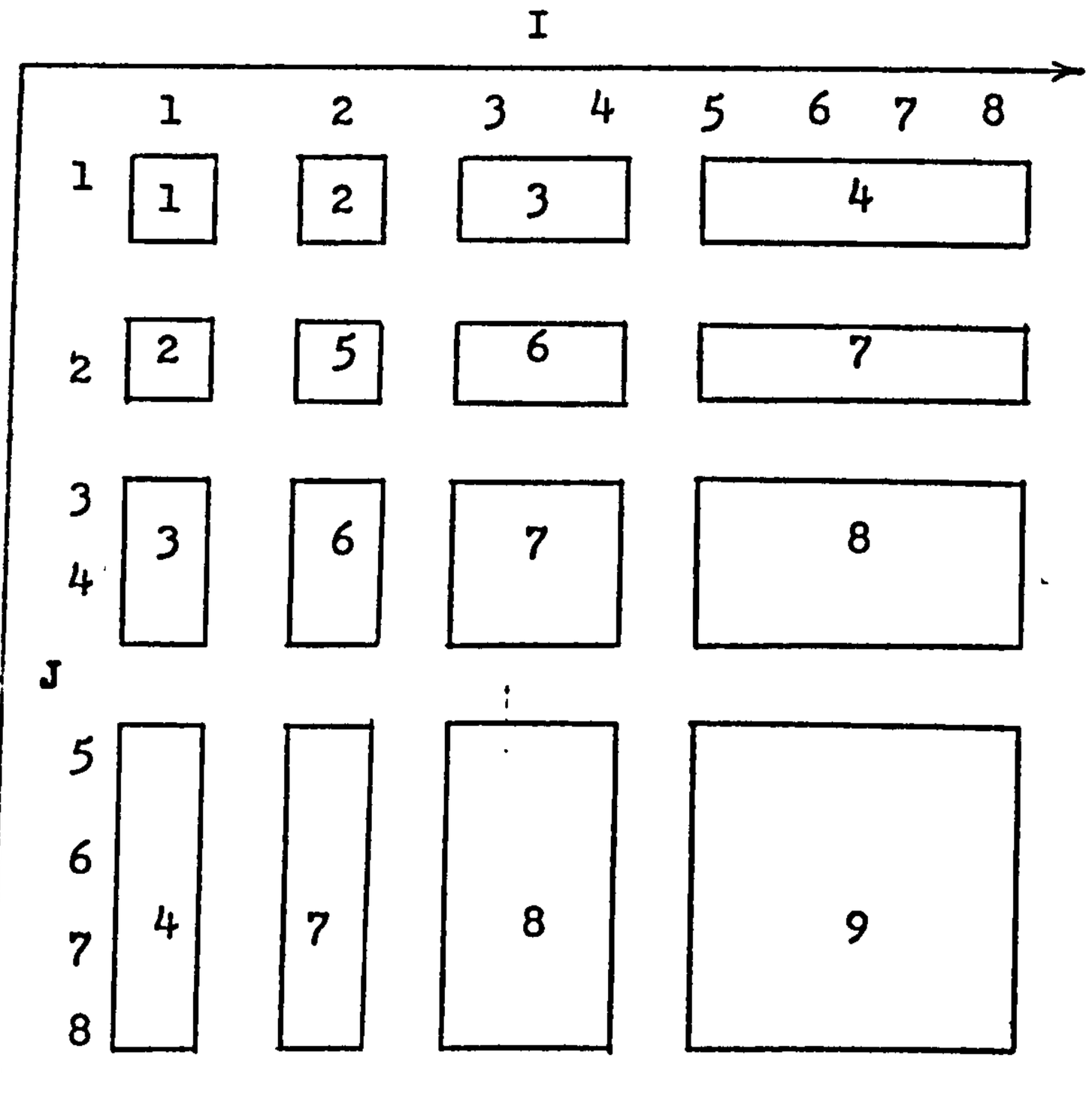


Figure (2.18). Map indicating the spatial positions of subblocks as obtained theoretically. The squares and rectangles represent subblocks, while the numbers inside indicate their orders.

0	36	45	43
32	40	47	47
37	43	48	49
36	42	47	49

(a) Image A

0	43	53	49
35	47	53	53
40	49	53	54
39	49	54	55

(b) Image B

0	35	44	42
37	44	48	50
41	45	47	51
37	41	44	48

(c) Image C

0	34	42	40
30	40	46	46
33	41	45	48
32	40	44	47

(d) Image D

Figure (2.19). Actual levels maps for a block transform size of 4x4. Values shown are relative to dc term and expressed in dB. The minus signs are dropped for clarity.

J=I	I=1	2	3	4	5	6	7	8
1	0	35	42	41	52	50	49	48
2	31	38	43	45	54	51	50	52
3	36	40	44	47	54	51	51	54
4	36	43	46	49	53	53	54	56
5	41	44	47	50	53	54	54	57
6	41	44	47	49	53	53	54	57
7	41	44	47	50	54	53	54	57
8	41	44	46	49	52	53	53	57

(a) Image A

I=	1	2	3	4	5	6	7	8
	0	39	50	45	58	58	56	51
	32	44	52	51	59	58	58	57
	37	46	52	52	59	59	58	58
	38	47	53	54	60	59	58	60
	44	50	54	55	59	59	59	60
	44	50	54	55	60	59	59	61
	43	51	54	56	59	59	60	61
	42	51	54	56	60	59	60	61

(b) Image B

J=								
1	0	33	39	39	50	47	45	46
2	34	45	46	49	54	53	52	56
3	41	46	50	51	54	54	56	57
4	39	47	49	50	52	54	55	57
5	45	49	51	50	52	52	55	57
6	44	50	51	51	53	54	56	59
7	45	48	50	52	53	53	55	58
8	40	43	44	46	48	49	50	55

(c) Image C

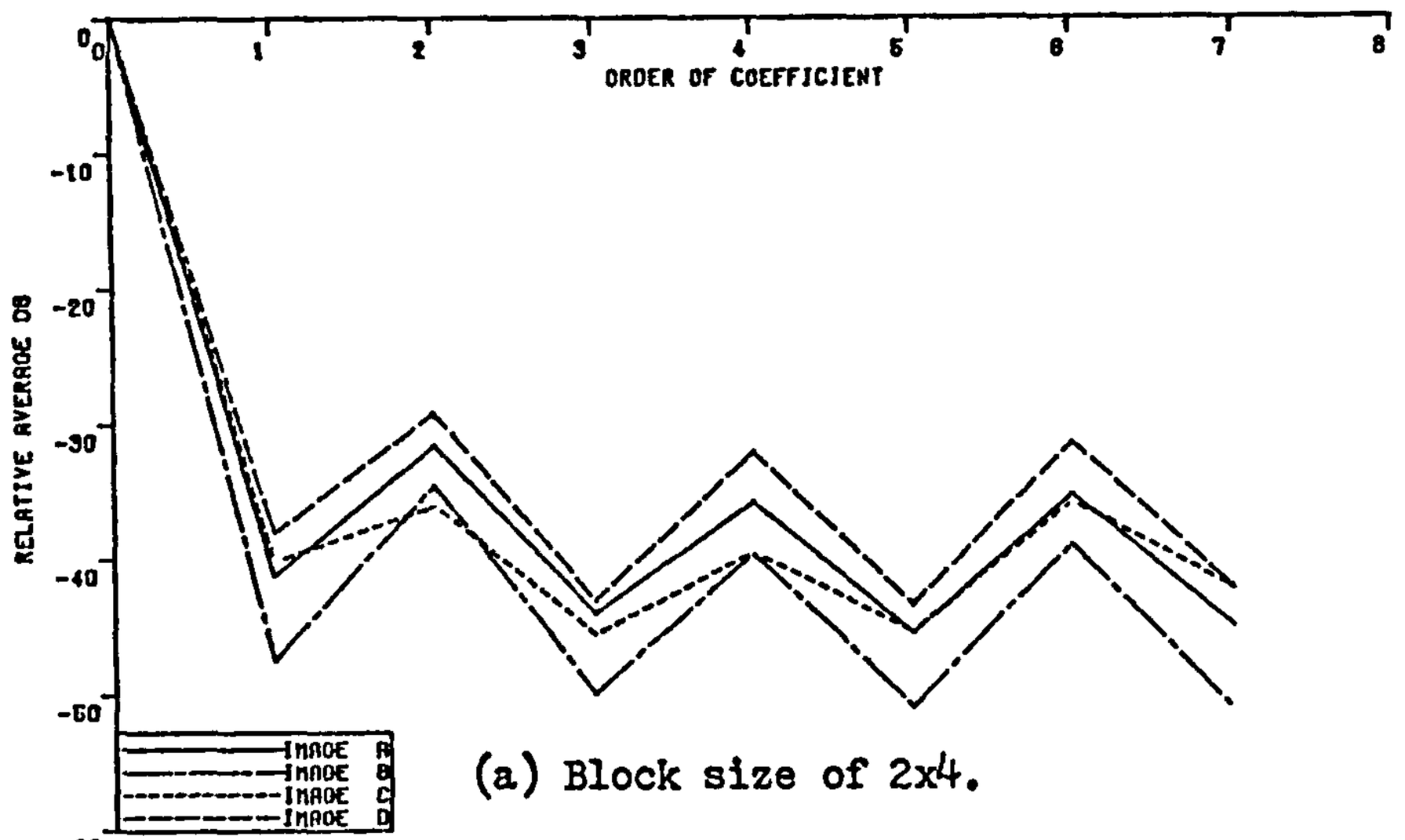
	0	30	37	37	47	45	44	43
	28	38	42	44	49	49	48	50
	32	40	44	46	52	51	50	53
	33	42	45	46	51	50	51	53
	36	42	46	47	51	51	52	54
	37	43	46	47	50	51	52	55
	36	43	45	47	50	50	52	54
	35	41	44	45	48	49	50	53

(d) Image D

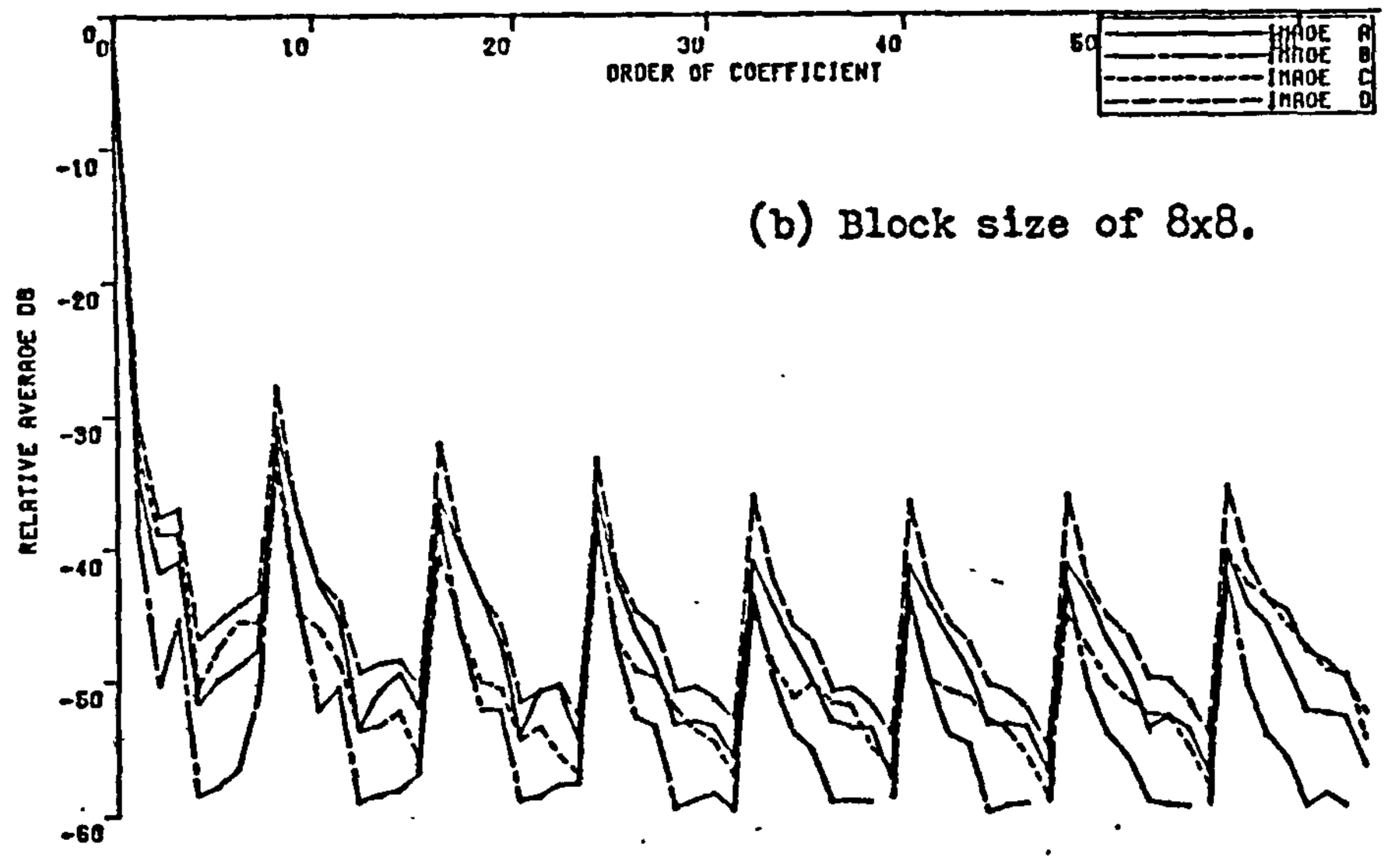
Figure (2.20). Actual levels maps for a block transform size of 8x8. Values shown are relative to dc term and expressed in dB. Minus signs are dropped for simplicity.

Figures (2.21). show graphs of actual relative averages of coefficients in different block sizes, and for different images.

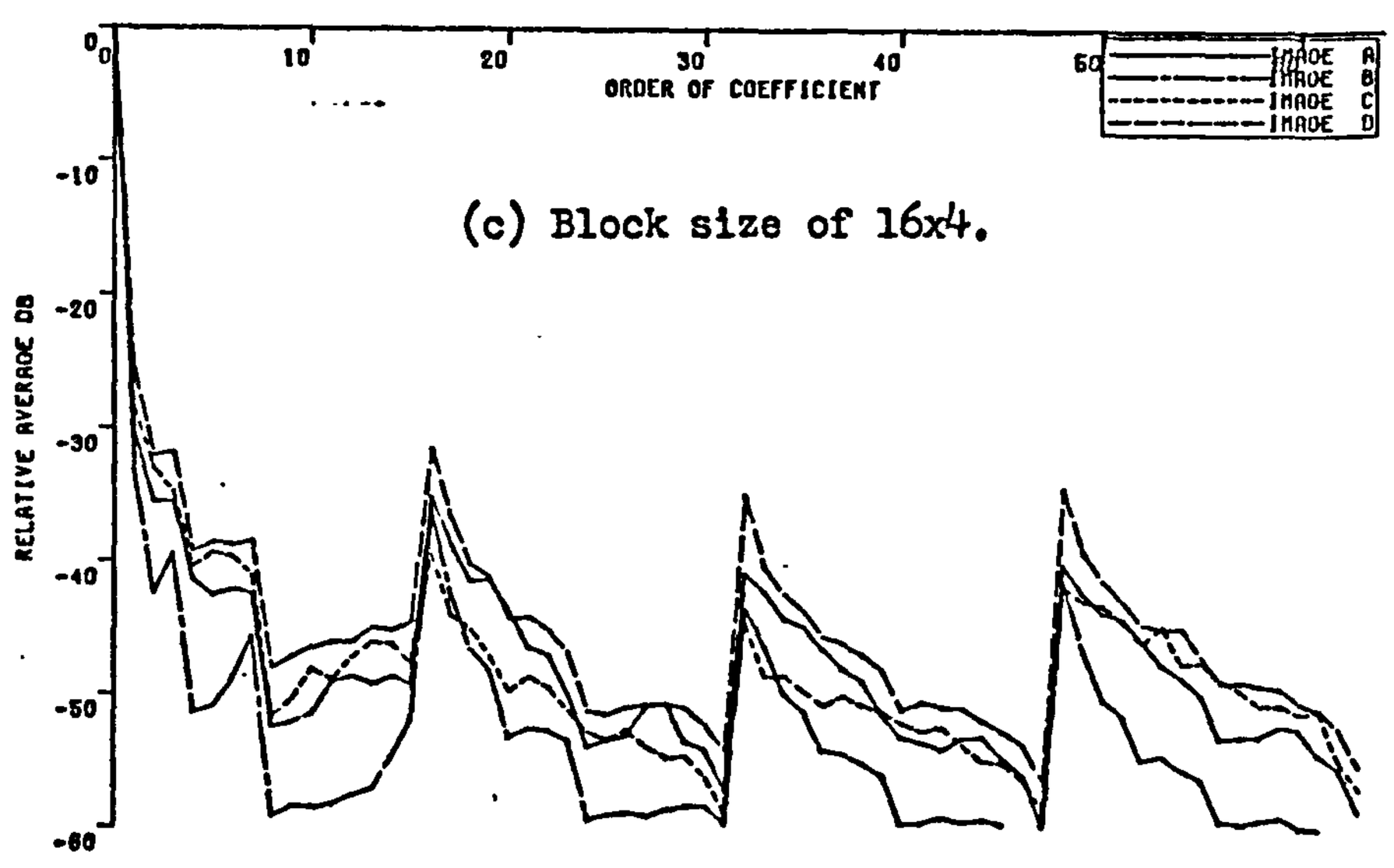
Figure (2.21). Relative averages of absolute values of coefficients in different block sizes nad different images.



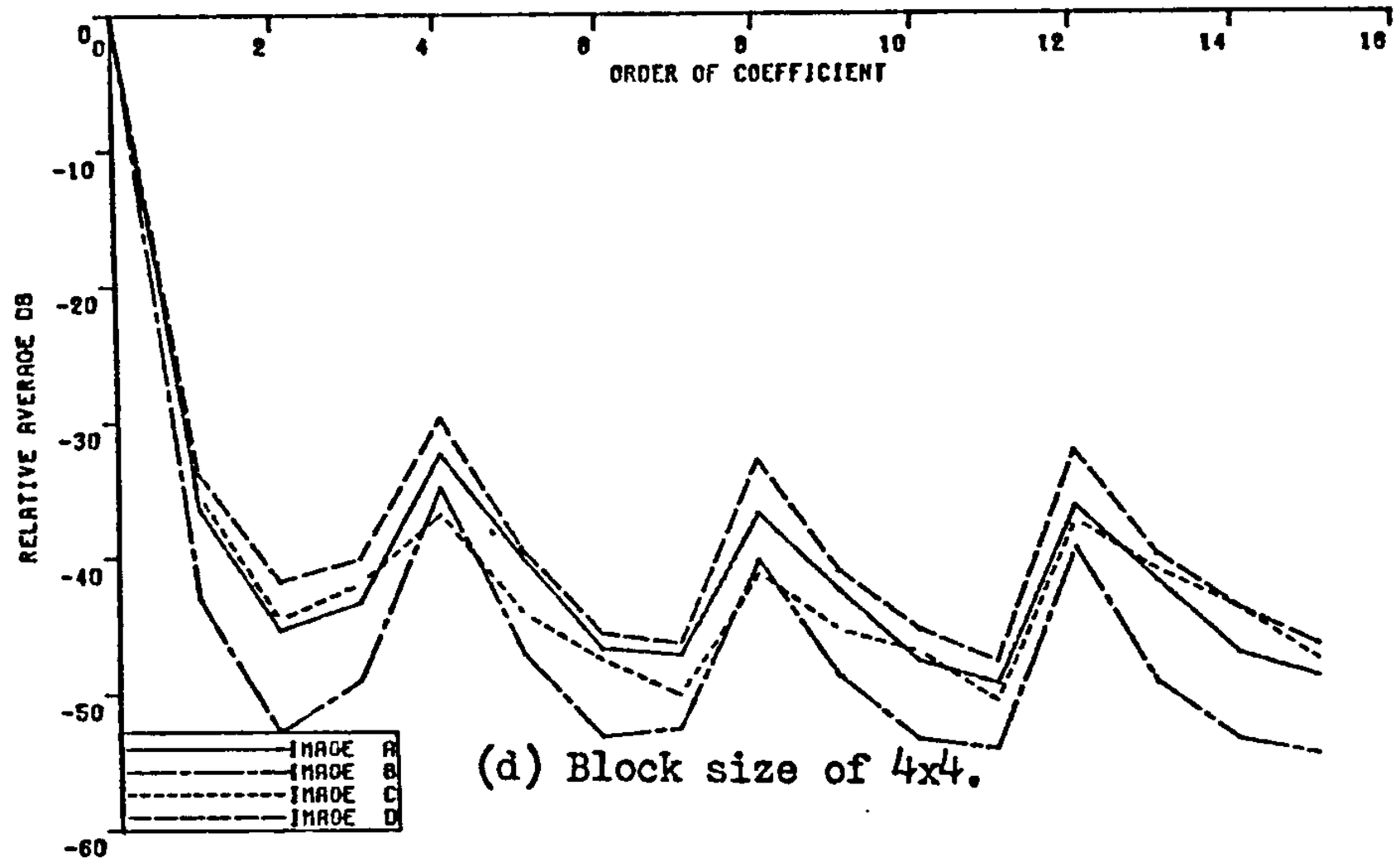
(a) Block size of 2x4.



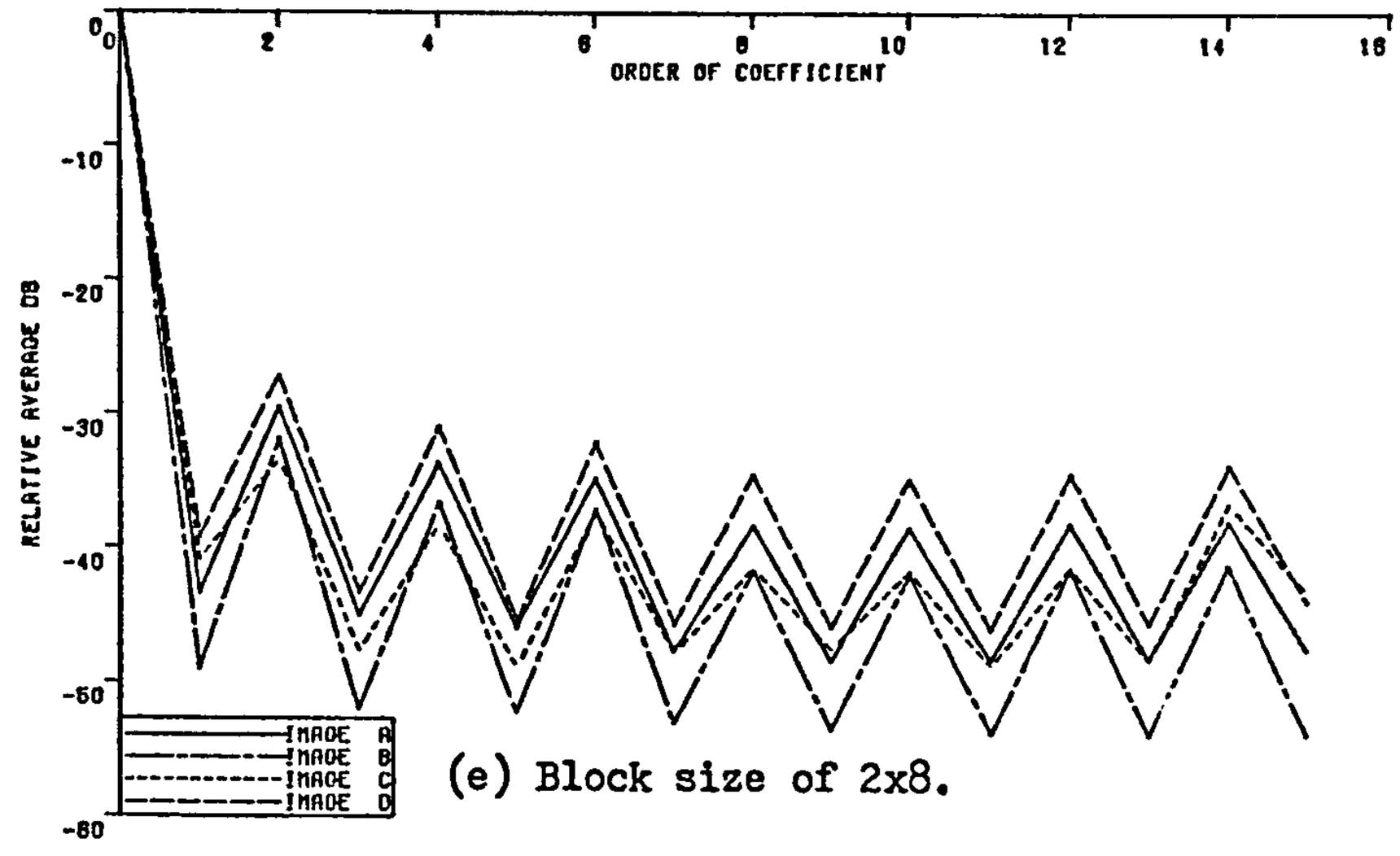
(b) Block size of 8x8.



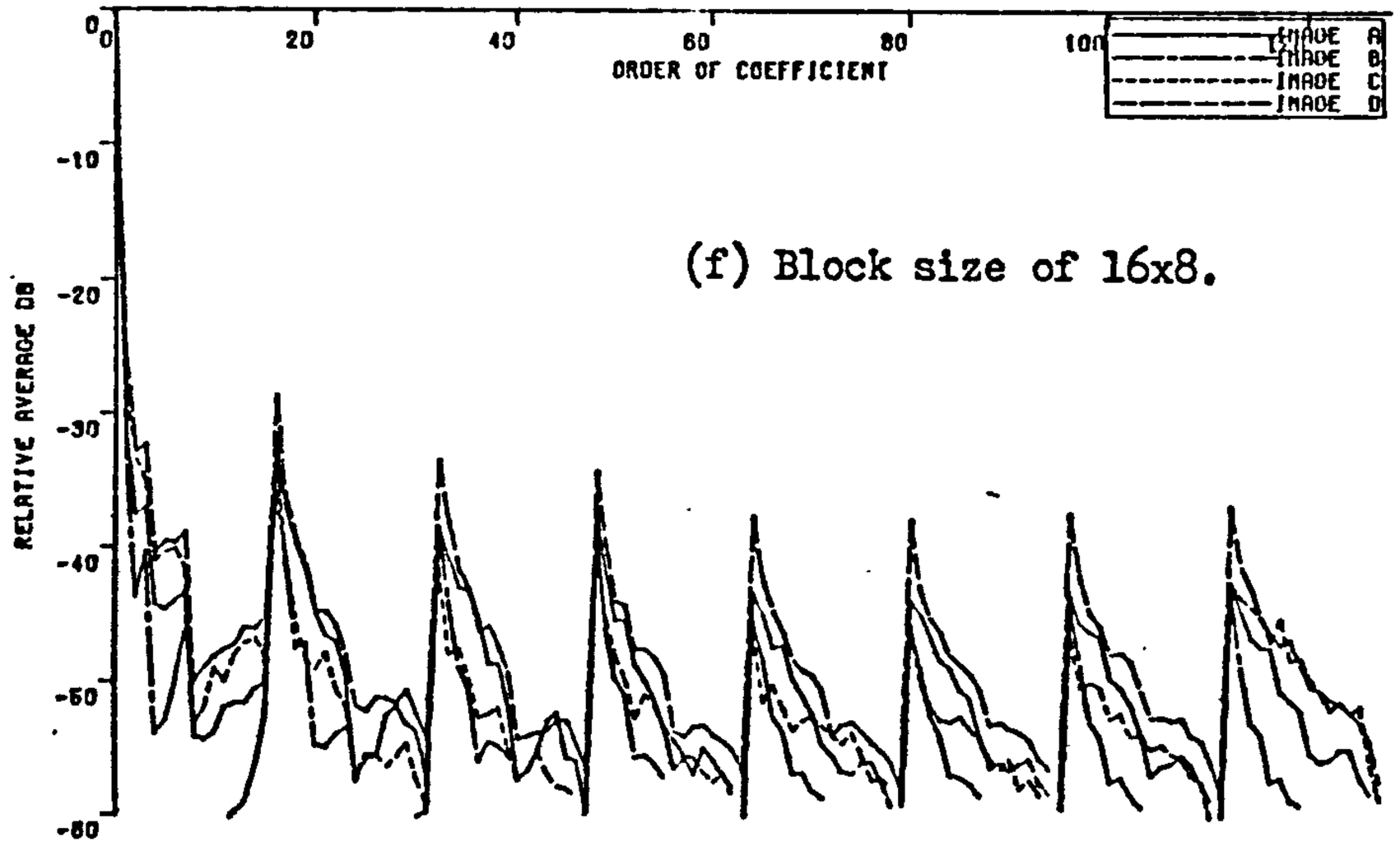
(c) Block size of 16x4.



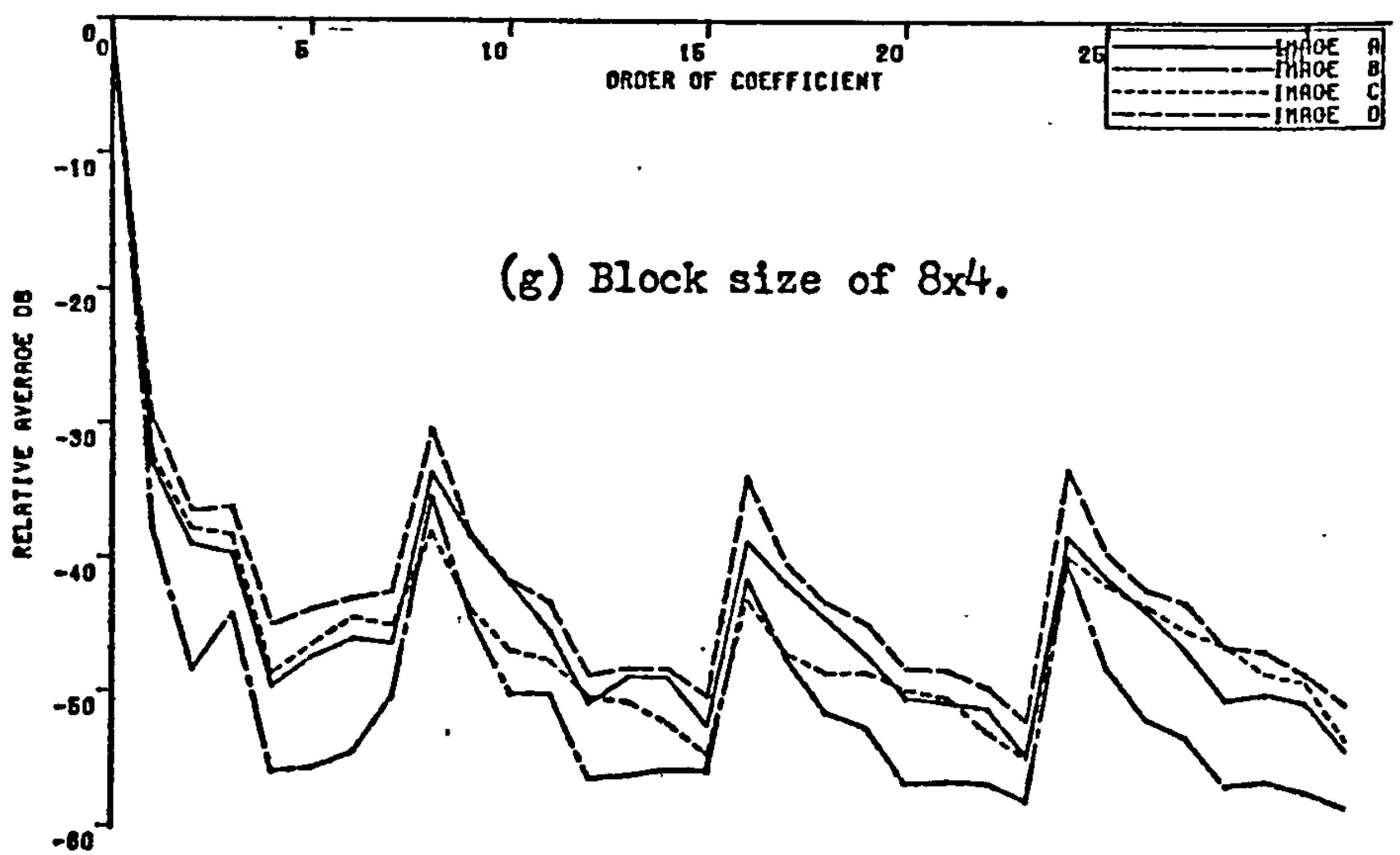
(d) Block size of 4x4.



(e) Block size of 2x8.



(f) Block size of 16x8.



(g) Block size of 8x4.

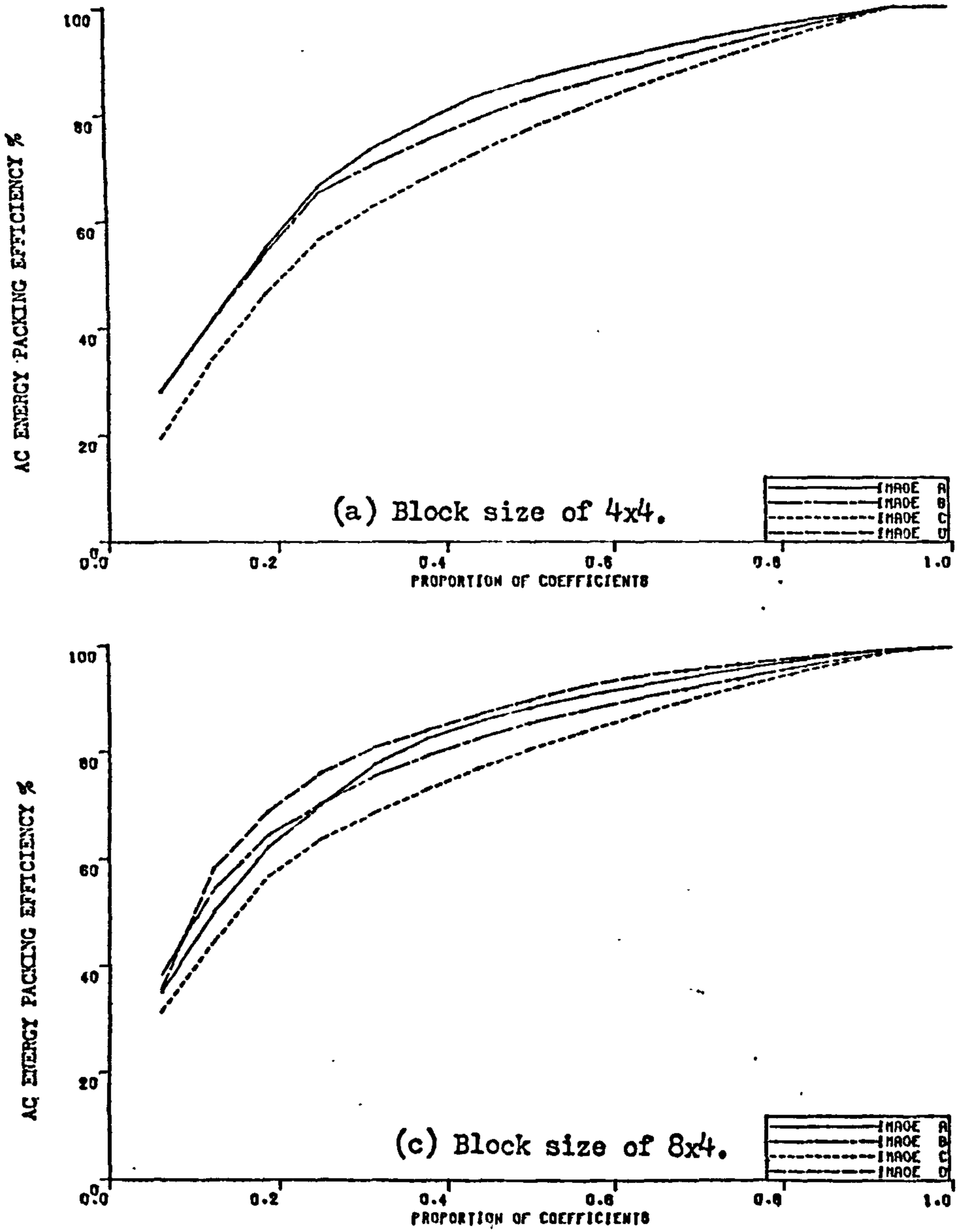
2.2.4 Energy Packing Efficiency:

The fact that there is no unique order for block transform coefficients will affect the calculation of energy packing efficiency. There are two possible ways of such a calculation. The first one is to assume the applicability of the theoretical order as obtained in Section (2.2.3). This will assign a fixed order to each coefficient regardless its actual value. The second, is to rearrange the coefficients of the block transform into a one dimensional array ordered in the descending values of averages, regardless of actual positions of coefficients in the transform levels map. Obviously, the packing efficiency calculated in the latter way will be more optimistic than the real value. However, to overcome the problem of nonuniqueness of order in block transform, the second method is used. For the values of efficiency obtained as a result of this to be more practical, a factor of about 0.8- 0.9 could be used for correction of efficiencies obtained (subject to particular statistical investigation).

Table (2.20) shows energy packing efficiencies for different block sizes and different image categories. Figures (2.22) show also the values of packing efficiencies and comparisons for different sizes and different categories. All values in Tables and Figures are not corrected. One important result which can be drawn from these tables and figures, is that, although the energy packing efficiency is computed in the more optimistic way, it is still generally lower than that of line transforms of equivalent transform sizes.

Table (2.20). Energy packing efficiency for a block size of 2x8, and for different categories.

Category	A		B		C		D	
Up to coefficient.	ac energy%	all energy%	ac energy%	all energy%	ac energy%	all energy%	ac energy%	all energy%
1	32.5	99.1	25.0	99.3	26.7	98.9	30.9	99.5
2	47.6	99.5	36.6	99.5	40.3	99.3	44.4	99.7
3	59.7	99.6	48.0	99.6	51.6	99.5	57.3	99.8
4	67.0	99.7	57.9	99.7	61.2	99.6	64.8	99.9
5	74.2	99.8	63.9	99.78	69.8	99.7	71.4	99.9
6	81.0	99.84	69.8	99.82	77.7	99.78	77.9	99.94
7	87.6	99.88	75.7	99.86	85.5	99.84	84.1	99.96
8	90.2	99.93	81.4	99.89	90.3	99.91	87.3	99.985
9	92.0	99.95	85.7	99.92	92.3	99.94	89.3	99.990
10	93.5	99.96	88.3	99.94	94.1	99.95	91.2	99.992
11	95.1	99.97	90.9	99.96	95.4	99.97	93.1	99.993
12	96.4	99.98	93.3	99.97	96.6	99.97	94.9	99.995
13	97.6	99.99	95.6	99.98	97.8	99.98	96.7	99.996
14	98.8	99.99	97.9	99.98	98.9	99.99	98.4	99.997
15	100	99.995	100	99.99	100	99.99	100	99.999
16	100	100	100	100	100	100	100	100



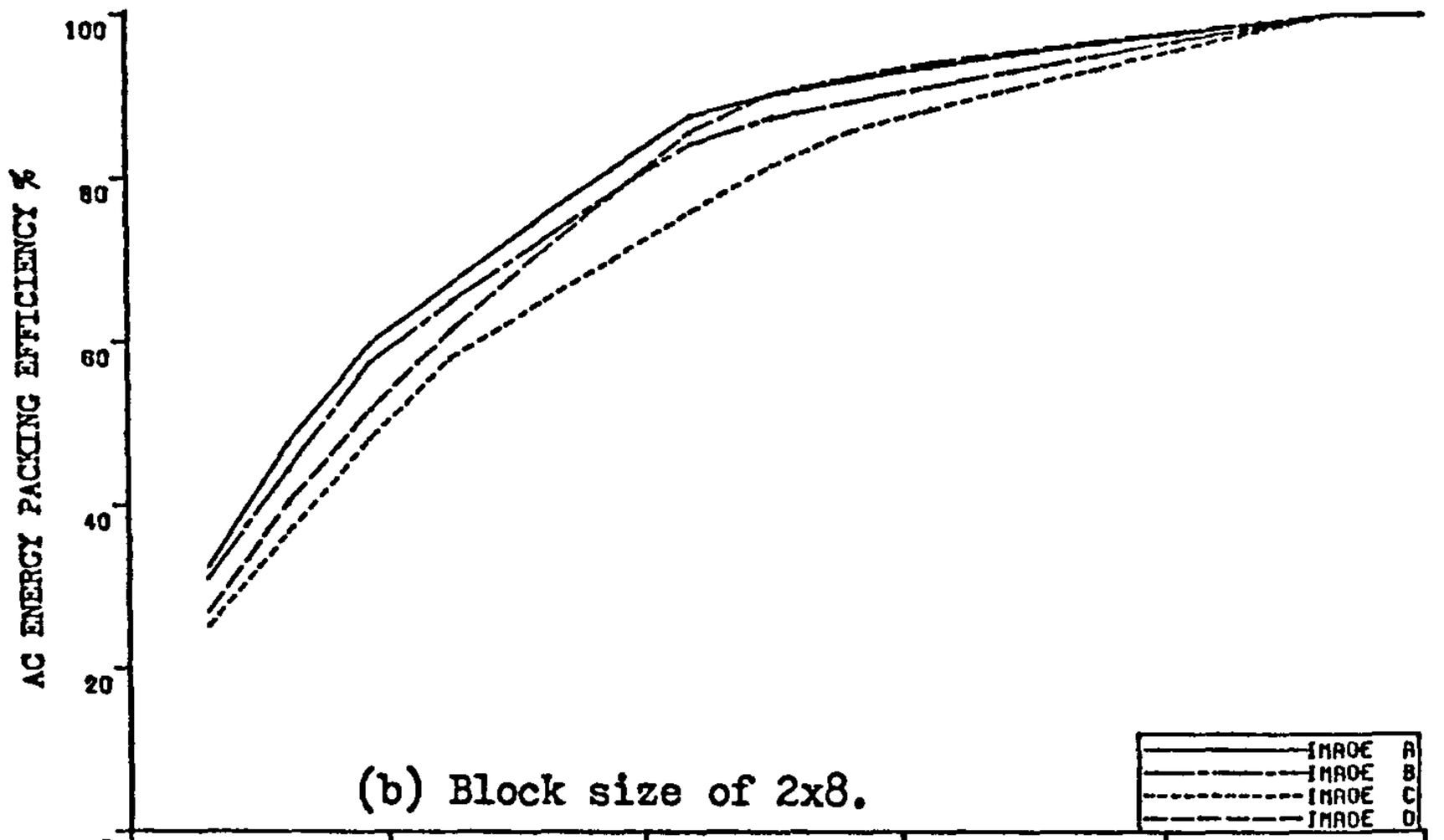
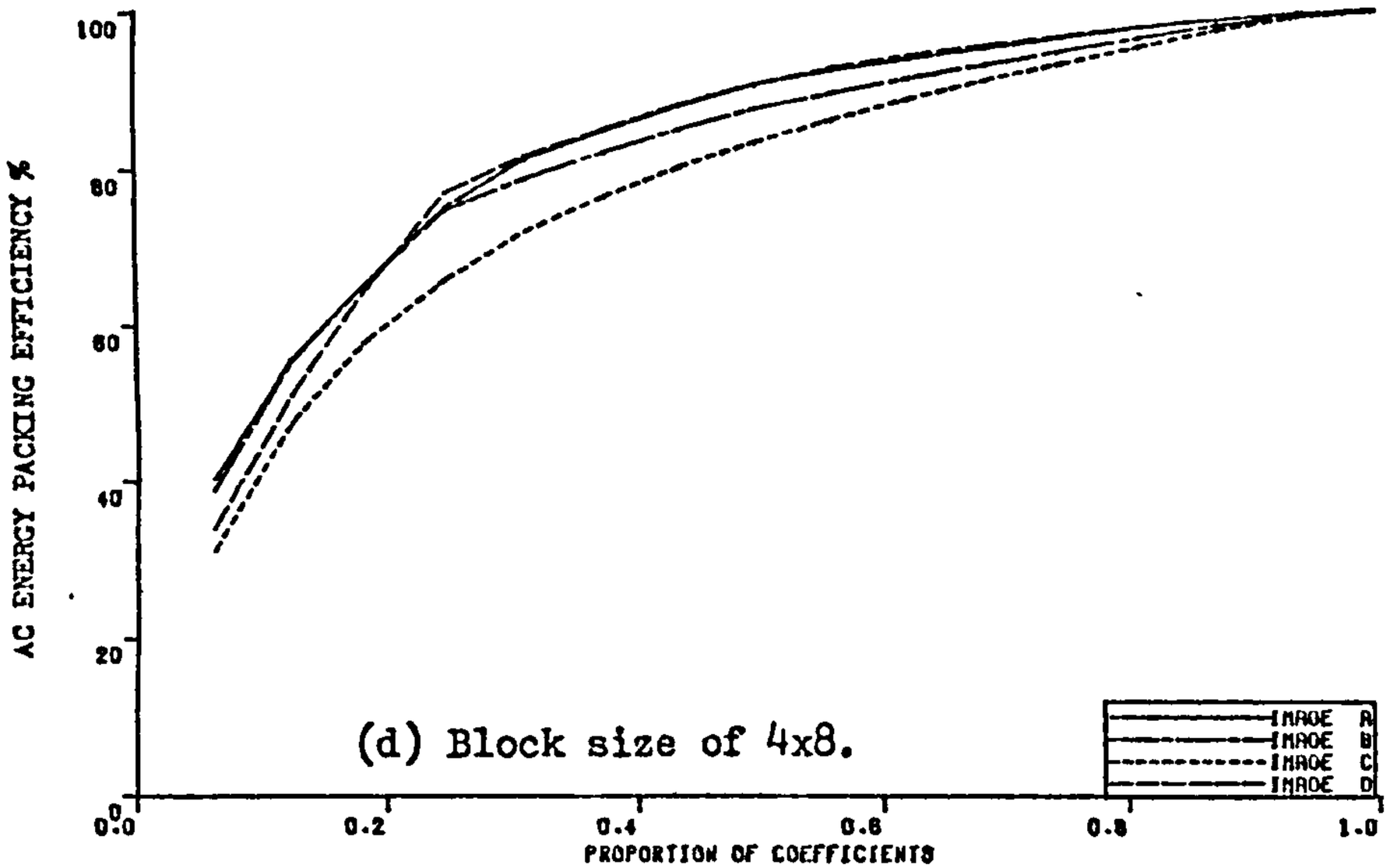
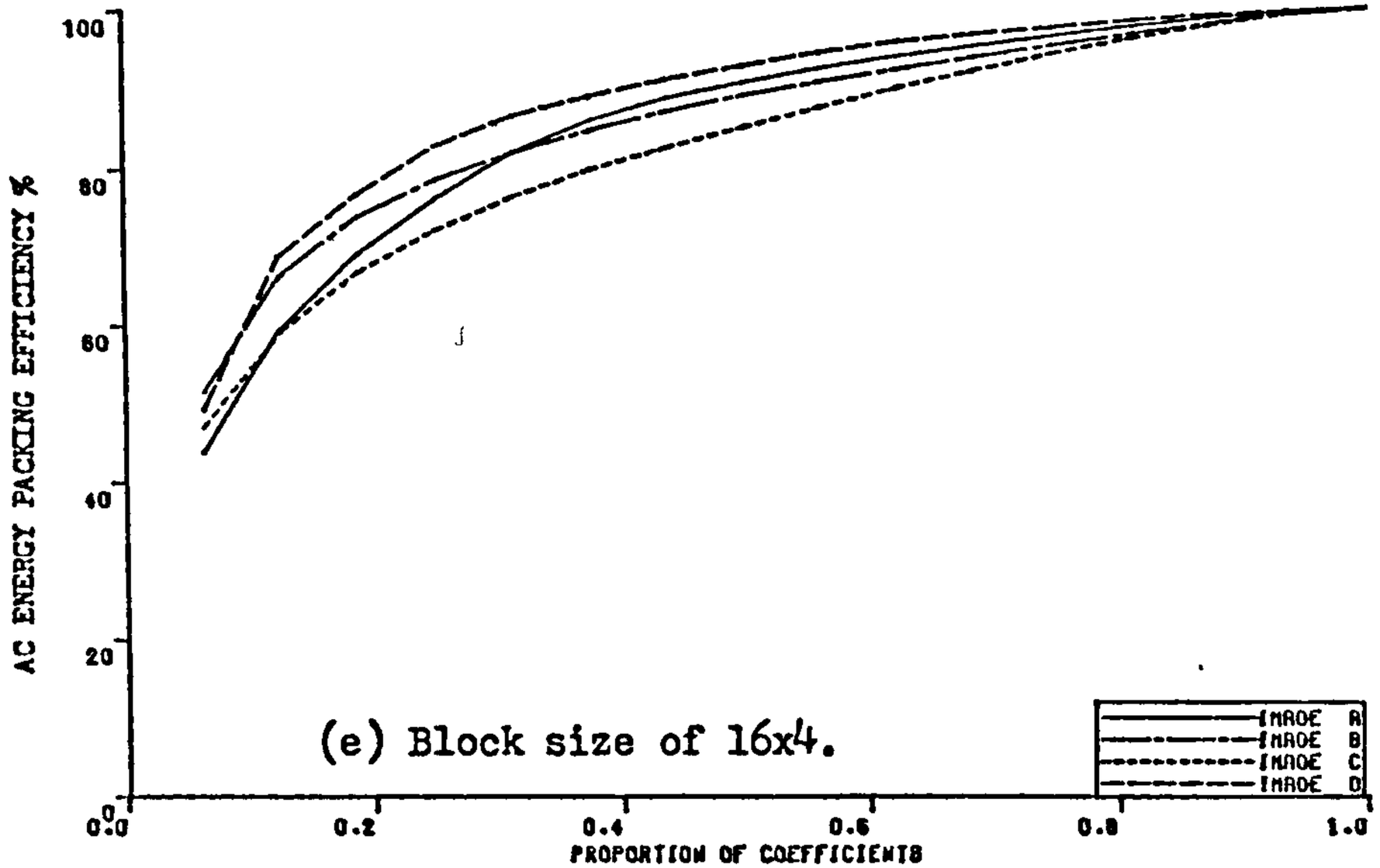


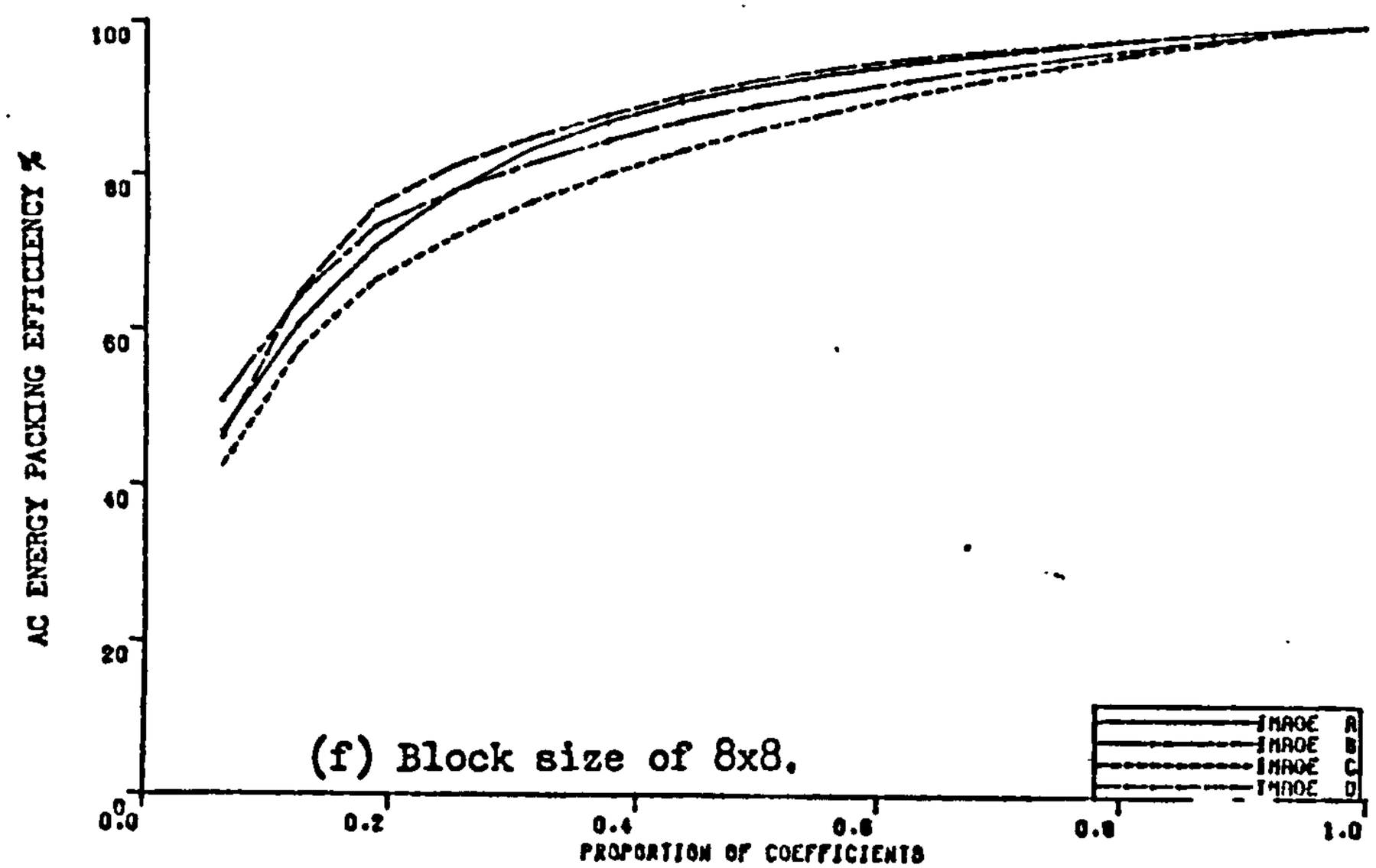
Figure (2.22).AC energy packing efficiency for different block sizes and different images.



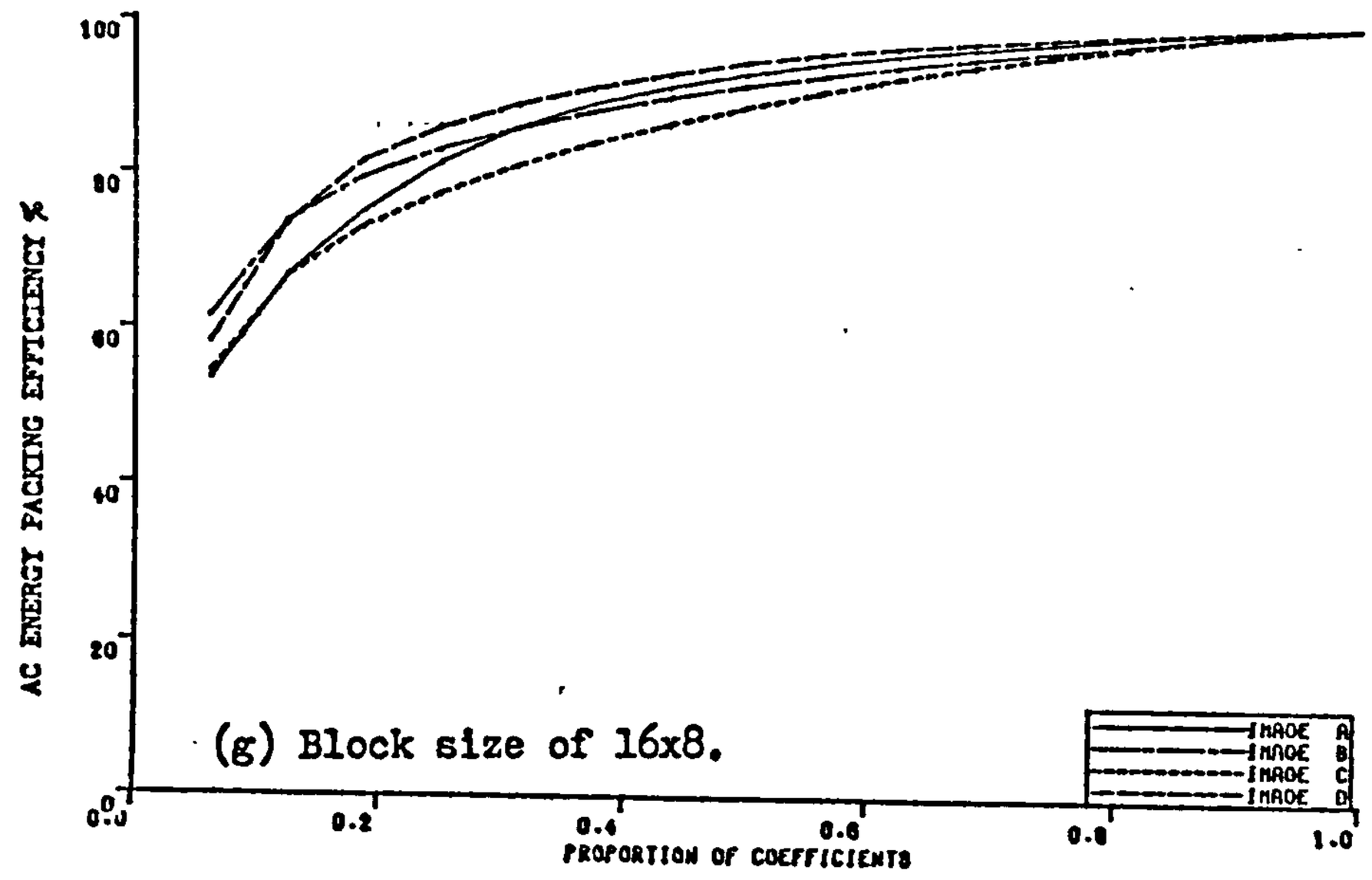
(d) Block size of 4x8.



(e) Block size of 16x4.



(f) Block size of 8x8.



(g) Block size of 16x8.

2.2.5. Abruptancy in Block Transform Domain:

As noticed from Sections (2.2.3. &2.2.4), the cocept of abruptancy in absolute values and energy packing efficiency is not so clear in case of block transform as in the line transform. Due to the nonunique divisions of block transform coefficients into subblocks, it is not possible to apply the measures set in Section (2.1.5), to the present case. However, as an example, if Figure (2.20) is carefuly examined, it will be found that, apart from the sharp decrease in values of ac coefficients relative to the dc term, relative values of different ac coefficients tend to change smoothly from one "theoretical" subblock to another.

2.2.6. Effects of Block Shapes:

In the foregoing sections, different block sizes were considered, Although the general block shape for all previous studies was a square one, no practical reason has been given for that. It may be only for simplicity of computations and block construction. In this present study, rectangular blocks with different dimensions were used. As the primary interest is in the energy packing efficiency, effects of these dimensions on that efficiency w:ll be discussed here. Figure (2.22) shows energy packing efficiency for different block shapes.From this figure, it is apparent that for the same size (i.e, same number of coefficients), there is, virtually, no difference between square and non-square blocks. Moreover, there neither are differences in cases of reversed dimensions, (e.g. 8x4 and 4x8 blocks). This means clearly that for any block size, it is better to increase the horizontal dimension and decrease the vertical as this means cosiderable savings in storage requirements for lines storing.

2.3.Comparison between Line and Block Transforms:

Comparing results of corresponding Sections on line and block transforms, a general conclusion can be drawn as follows:

1. Block transformation needs more storage than line transform, in order to store a number of TV lines for building up the blocks. This storage requirement will increase sharply as the vertical dimension of the block increases.

2. The general characteristics of transform coefficients in both domains are virtually the same, regarding the probability density function and ranges of values.

3. Although the ac energy packing efficiency seems to be the same in both cases, figures and values in block transform case are higher than actual values due to the 'generous' ordering of coefficients as discussed in Section (2.2.4). If the correction factor is applied as suggested, much more lower values will be obtained. This is also clear from results in Section (2.2.5.), which shows that the principles of abruptancy is not applicable in block transform.

4. As a result of these comparisons, and cosidering the substantial storage requirements as well as the efforts in arranging and computation, it is concluded that the line transform is more efficient than block transform.

CHAPTER THREE

TRANSFORM DOMAIN CHARACTERISTICS OF COLOUR SIGNALS

Investigations of the characteristics of transform domain coefficients presented in Chapter 2 are extended here to colour signals. Due to the presence of chrominance subcarrier in this case, different ways of applying the transformation to the spatial samples are discussed. Characteristics for each case are studied with regard to their effects on bit rate reduction. As the values of samples will be directly dependent on the ratio between sampling frequency and subcarrier, the effects of changing this ratio are also discussed for two representative values, $3f_{sc}$, and $4f_{sc}$ sampling frequency. Extensive comparisons among all cases considered are also presented, as well as comparisons with monochrome signals.

3.0.1. Introduction

As in Chapter Two, the analyses in this Chapter were performed on stationary images. The Red/Blue/Green signals outputs from a Rank-Cintel MK4 (flat faced CRT) slide scanner (with signal to noise ratio of 50 dB) were coded into a composite video signal in an RGB coder. The composite video signal was clamped and passed into an 8-bits/sample, high speed video Analogue to Digital Converter ADC, (Micro Consultants AN- DI RAD C, 8 bits/sample, 15 MHz). The input signal to the ADC was clamped (in the range -0.5v to 0.5v).(63)

Three test images were used. These are categorised as

E: PORTRAIT.

F: LAKE DISTRICT VILLAGE.

G: PARROT.

However, due to some difficulties in reproducing colour images, only the monochrome version of one of these test images is shown in Figure (3.1).

Figure (3.1.). Monochrome version of test image E,
'PORTRAIT'.

3.0.2.Transform Application on Colour Signals:

In applying the principles of spatial transform to colour signals, special arrangements should be taken to account for the unique composition of these signals. As the colour signal is composed of three signals mixed together in a special manner, the way of transform application has to be altered. Several ways of transformapplication are possible. These can be broadly divided into two main categories:

1. Composite signal transform.

2. Components transform.

In the composite signal transform, the colour signals are dealt with in their composite form, as the name implies. This could be further divided into two subcategories. In the first, a data vector is composed of a number of succesive samples taken directly from the spatial domain. This will be called " Direct Composite Transform", or simply Direct transform. In the second subcategory, a data vector is composed from samples

a subcarrier interval apart from each other. This one will be called " Laced Composite Transform", or simply Laced Transform.

In components transform, the three components of a colour signal, luminance and chrominance components, are transformed separately in a direct manner. This could be thought of as plane transformation.(65)

Due to the presence of chrominance subcarrier, the characteristics of transform coefficients of sampled data will strongly depend on the relation between sampling frequency and subcarrier frequency. To study the effects of sampling frequency on transform domain coefficients, and with one eye on the bandwidth requirements, two sampling rates in common use are investigated here. These are the thrice- and four times subcarrier frequency.

3.1.Characteristics of Transform Coefficients in case of $3f_{sc}$ Sampling:

The instantaneous voltage 'e' of a complete PAL colour signal is given by(67);

$$e = E + U \sin wt \pm V \cos wt$$

where E is the luminance component,

U and V are the chrominance components,

w is the subcarrier angular frequency, and

$\pm$ refers to the alternate line switch, with the + sign corresponding to the line with colour burst at 135^{o}.

For a sampling frequency of thrice the subcarrier frequency, $3f_{sc}$, sampling phases could be chosen at wt $= \pi/2, -5\pi/6$, and $-\pi/6$ for the + lines. Thus, the three different sample values in one subcarrier interval will be: (68)

$$A = E + U \qquad \text{at wt} = \pi/2 \quad \ldots\ldots\ldots\ldots (3.1)$$

$$B = E - .5U - .866V \qquad \text{at wt} = -5\pi/6 \quad \ldots\ldots\ldots (3.2)$$

$$C = E - .5U + .866V \qquad \text{at wt} = -\pi/6 \quad \ldots\ldots\ldots (3.3)$$

Values for the alternate line can be found similarly.

The first general feature to be noted here is that, unlike the case of monochrome, values of consecutive samples differ from each other depending on the values of both U and V components.

3.1.1. Direct Transform at $3f_{sc.}$:

In this transform, consecutive samples along the TV line are grouped into vectors, and transformed as in the case of monochrome signals. From Equations (3.1-3.3), values of adjacent samples will vary from each other depending on values of chrominance components u and v.This means that the redundancy inherent in monochrome signals will not be met here.

3.1.1.1. Coefficient: Values and Variances:

Figure (3.2.).shows the relative values of averages for different vector sizes and three image categories. Tables (3.1- 3.3) show levels of coefficients values. From these tables and graphs, the following notes are apparent:

1. Unlike monochrome signals, valuesof coefficients are relatively higher, hence the number of coefficients which could be discarded will be relatively small. From the tables of levels, relative values do not go beyond -35 dB (in cases of vector sizes of N= 16 and 32), compared with -50 dB in the corresponding cases of monochrome signals.

2. Variations in coefficient values among image categories are much less spread than monochrome. General shapes of curves for all images are nearly similar and changes are in the range of only 2 dB.

3. The most noticeable difference than monochrome signals is the behaviour of coefficients values at medium and high sequencies. In the case of N= 16, coefficient H6 has higher value than adjacent ones, and the group of coefficients (H10- H12) have even higher values. In the case of N=32, this is noticed at H11 and (H_{18}- H_{26}) respectively. This applies to all image categories. This fact could be explained by the structure of composite signal. As mentioned in Section (3.1.1.), adjacent samples in one vector will have different values from each other. As ac Hadamard coefficients are the differences between sample values, then high sequency

coefficients, which represent differences between closely adjacent samples, will therefore be higher than in the case of monochrome.

4. From Tables (3.1- 3.3), all coefficients relative averages are above the -42 dB level, while in monochrome, a cosiderable proportion was beyond that level.

As for the variances of coefficients , which indicate the degree of spreading values, Figure (3.3) shows their relative values for some vector sizes. From figure, variances change nearly as values of averages. Again, H_6 in a vector of N= 16, has nearly more than double the value of its neighbours' ,where the group of coefficients (H_{10}- H_{12}) have large values, particularly H_{11}.Changes of variances with image categories are slight.

Table (3.1). Levels of Direct transform coefficients. N=16.

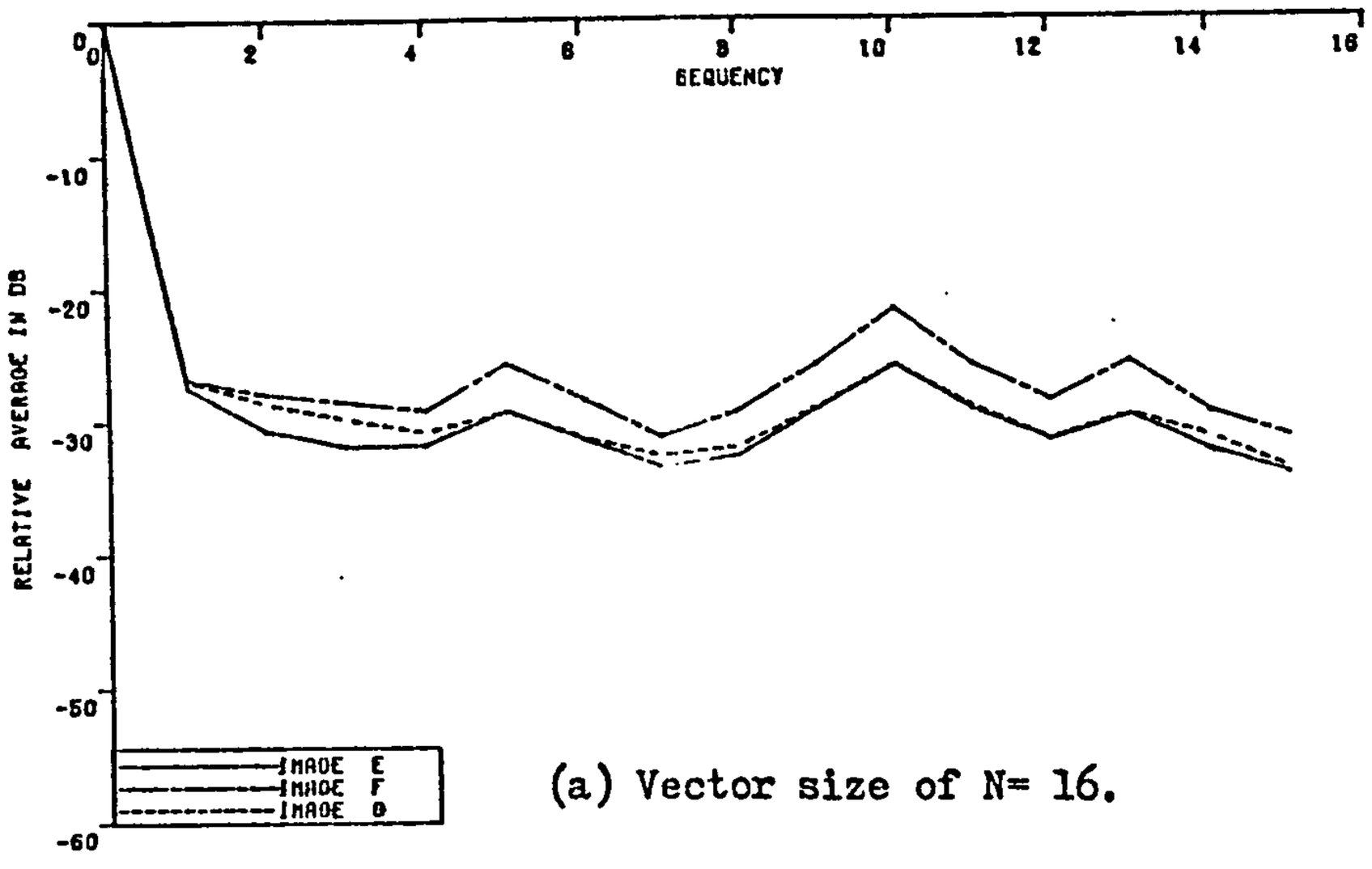
Levels of decrease than dc term in dB.		Percent of coefficients lying between these levels.		
From	To	E	F	G
18	24	-	6.3	-
24	30	12.5	37.5	12.5
30	36	68.8	50	68.8
36	42	12.5	-	12.5
42	48	-	-	-

Table (3.2). Levels of Direct transform coefficients. N= 32.

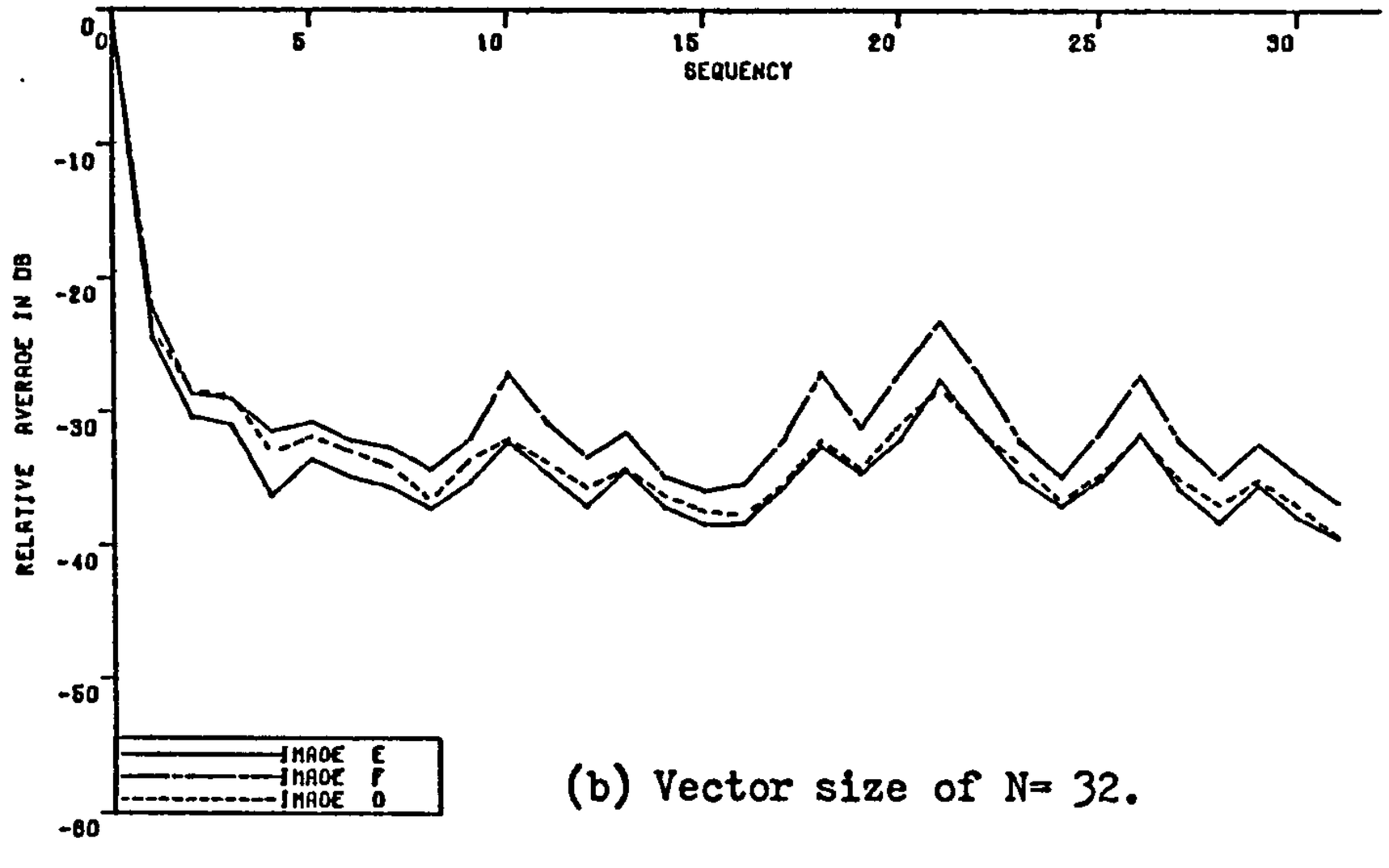
Levels of decrease than dc term in dB		Percent of coefficients lying between these levels.		
From	To	E	F	G
18	24	-	3.13	-
24	30	6.3	21.9	9.4
30	36	43.8	56.3	46.9
36	42	43.8	15.6	37.5
42	48	3.13	-	3.13
48	54	-	-	-

Table (3.3). Levels of Direct transform coefficients. N=64.

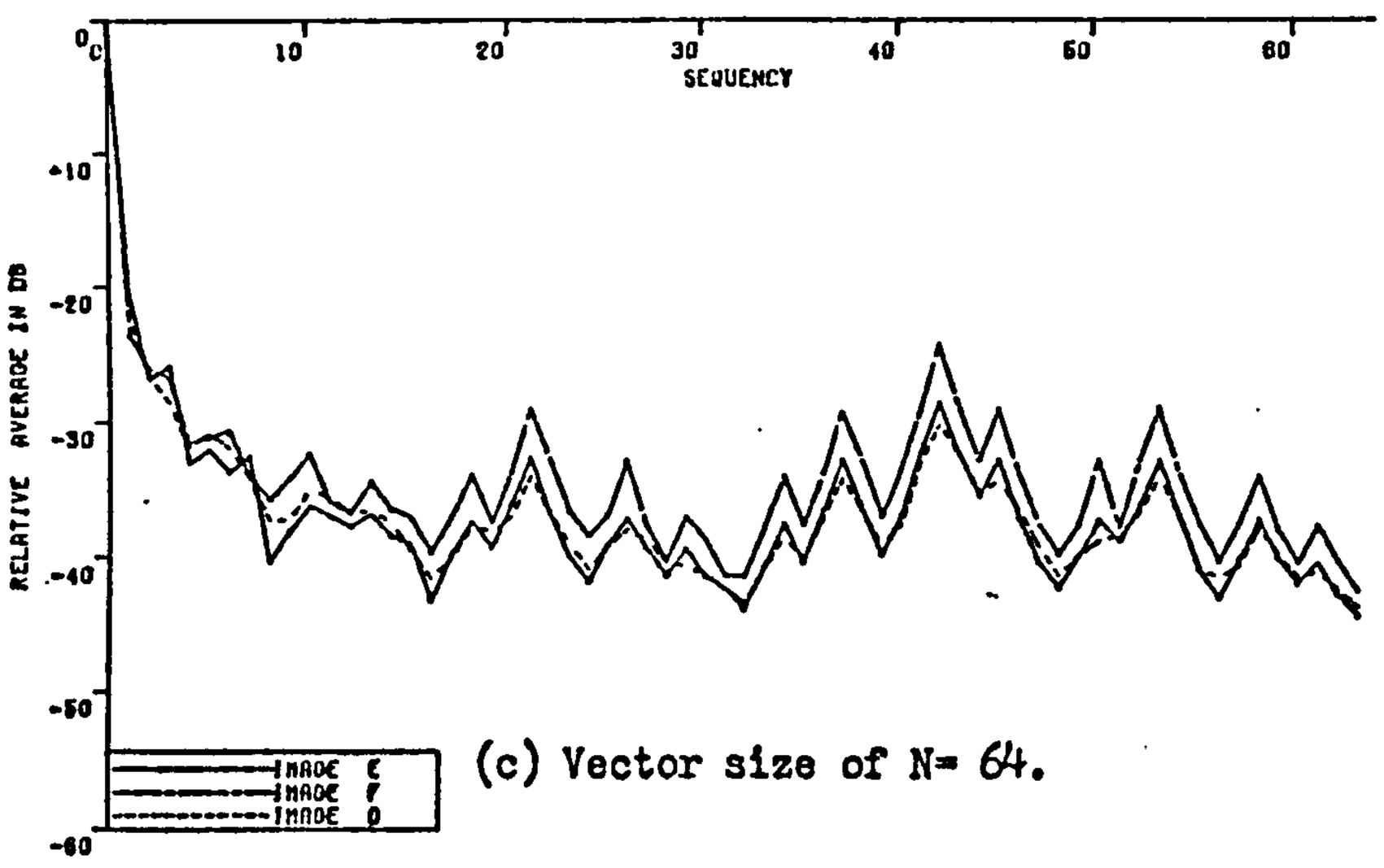
Levels of decrease than dc term in dB.		Percent of coefficients lying between these levels.		
From	To	E	F	G
18	24	1.6	-	1.6
24	30	4.7	15.6	4.7
30	36	17.2	34.4	18.8
36	42	57.8	45.3	59.4
42	48	17.3	3.1	14.1
48	54	-	-	-



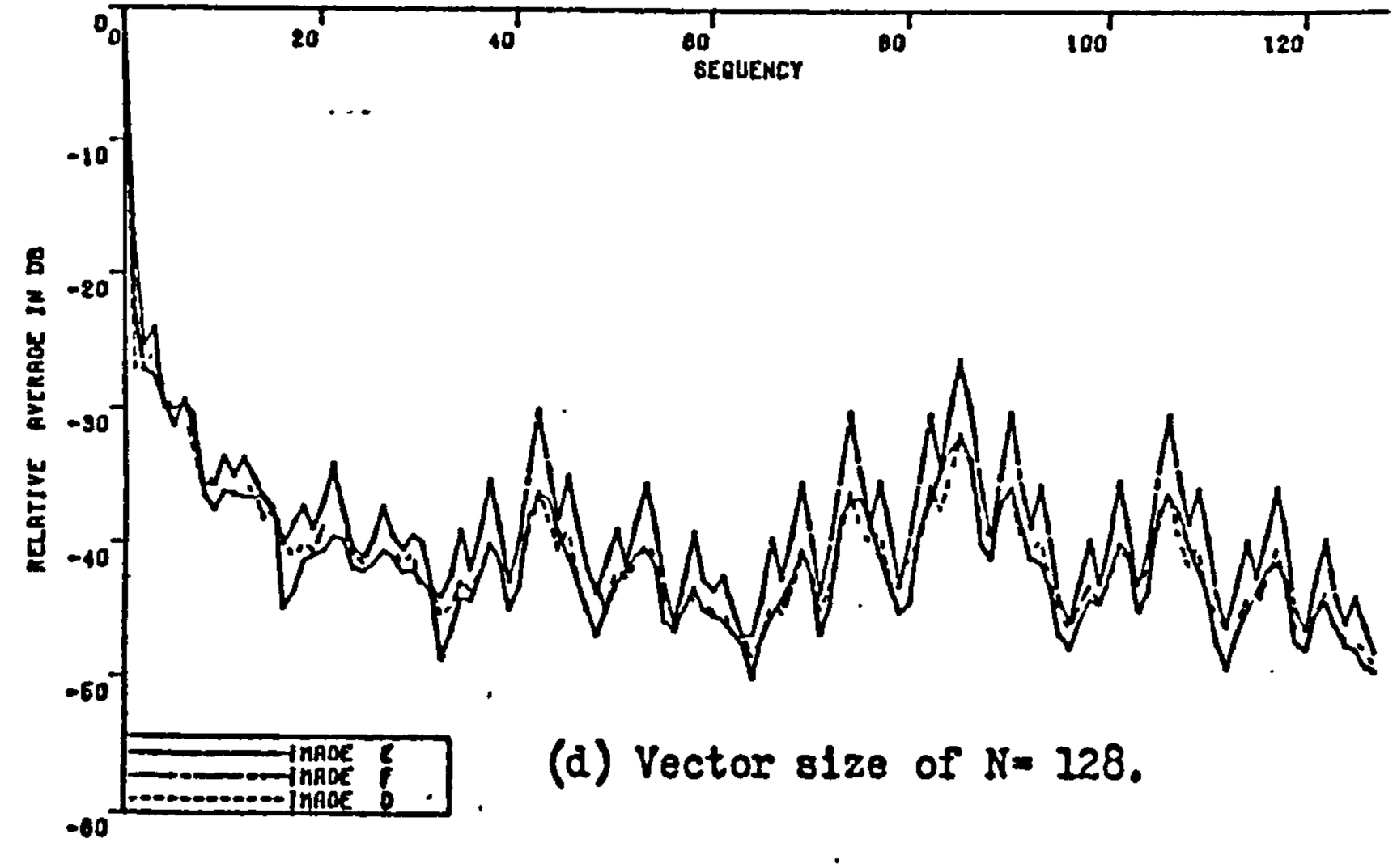
(a) Vector size of N= 16.



(b) Vector size of N= 32.

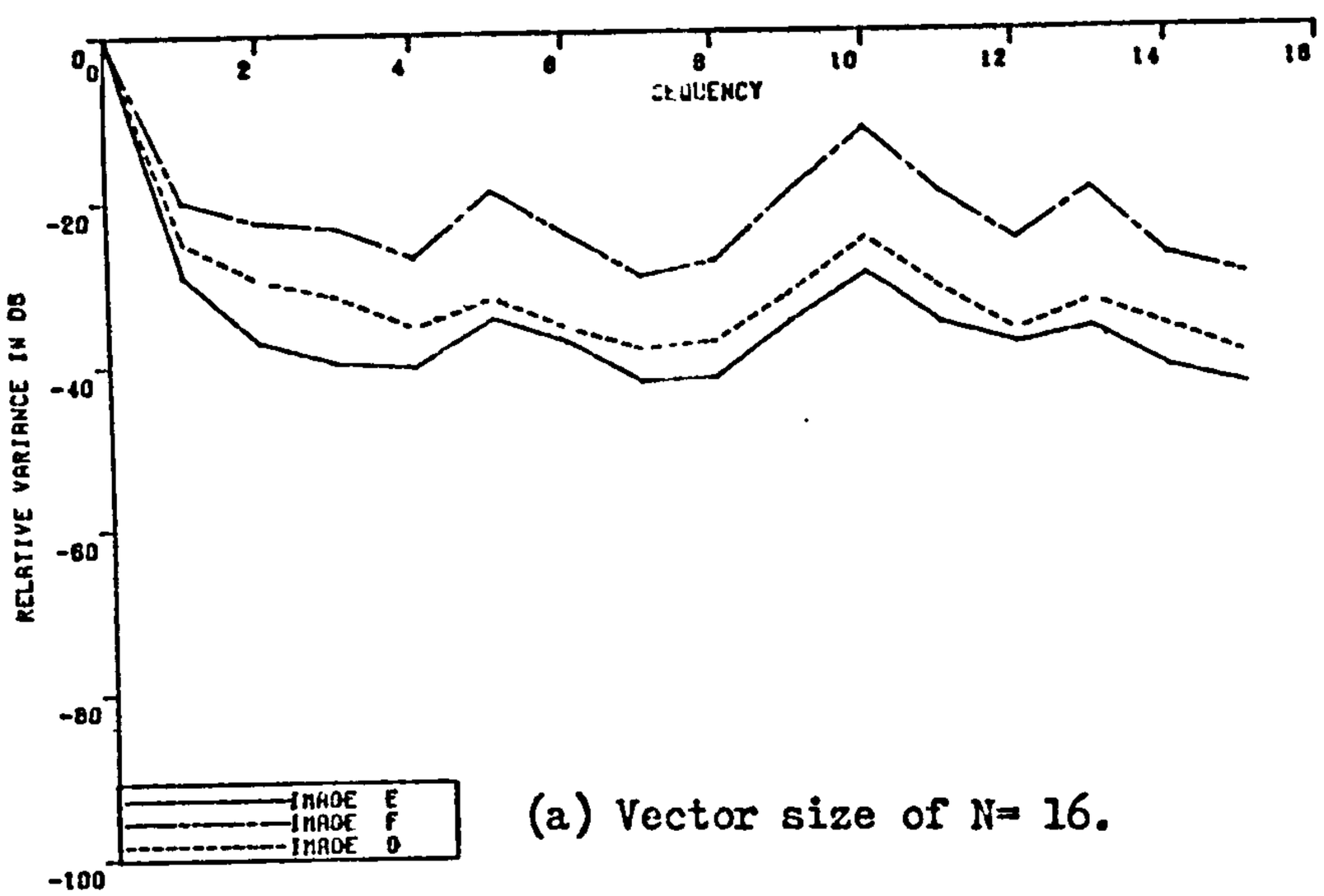


(c) Vector size of N= 64.

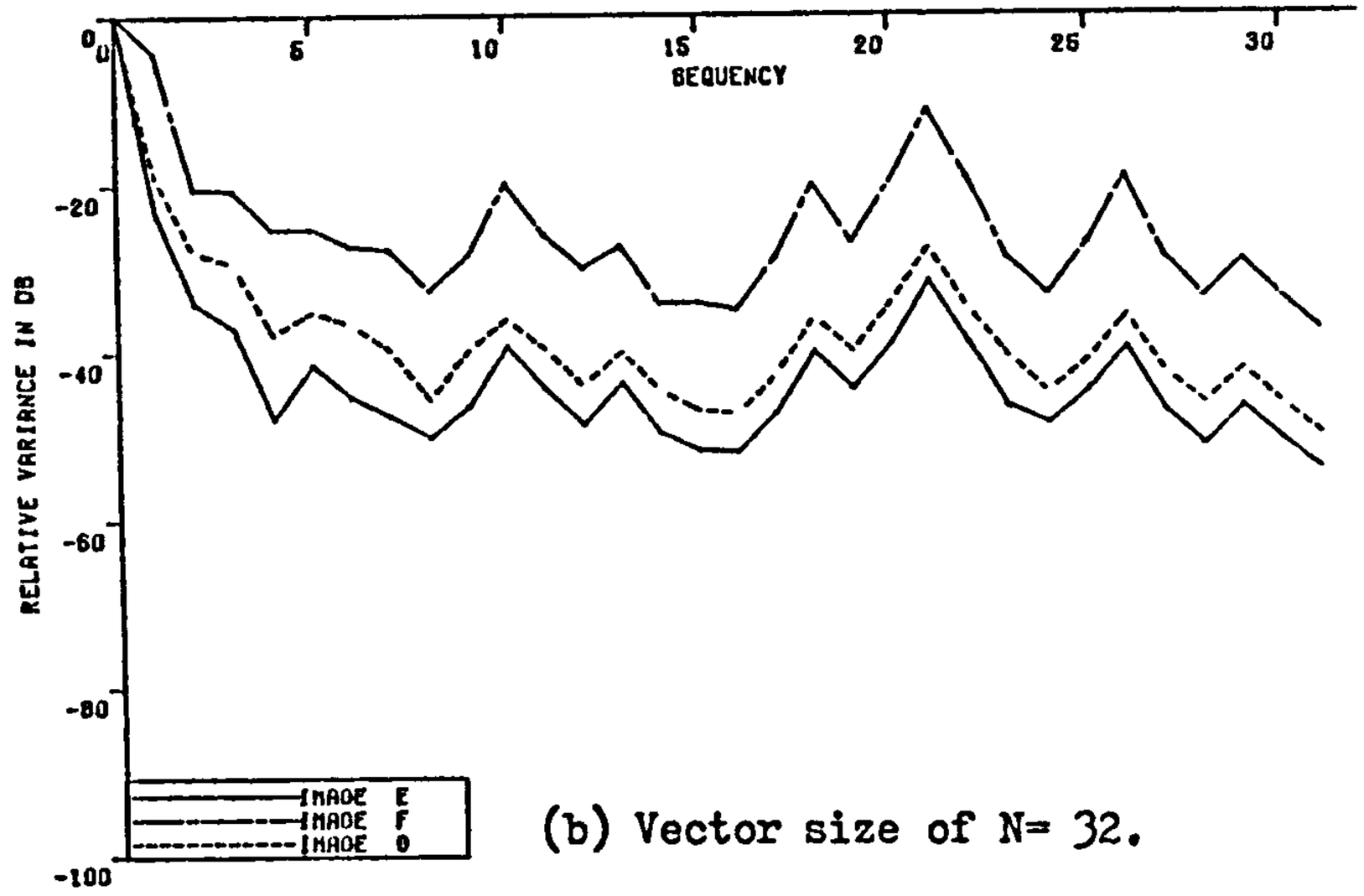


(d) Vector size of N= 128.

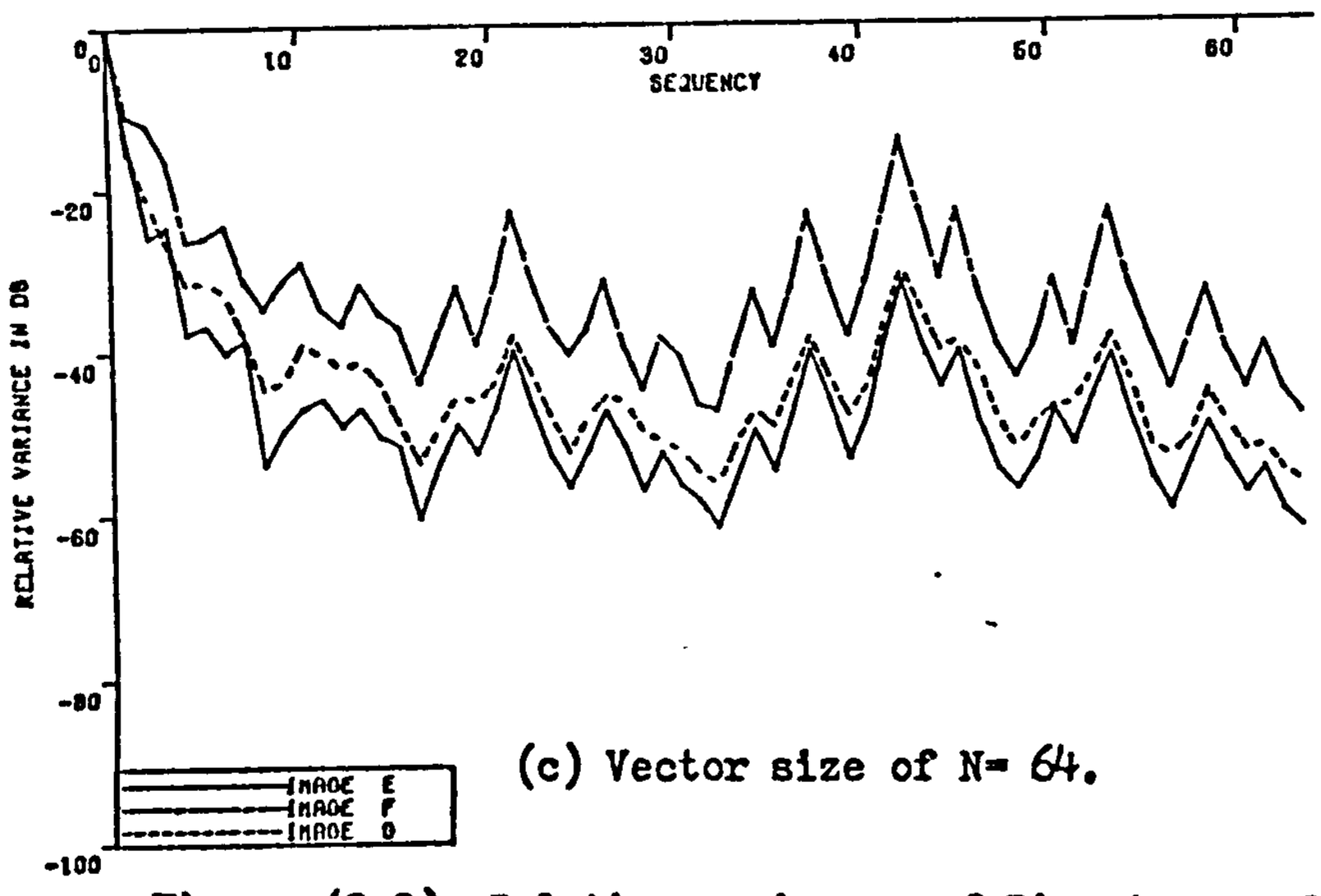
Figure (3.2). Relative averages of transform coefficient absolute values for Direct transform at $3f_{sc}$.



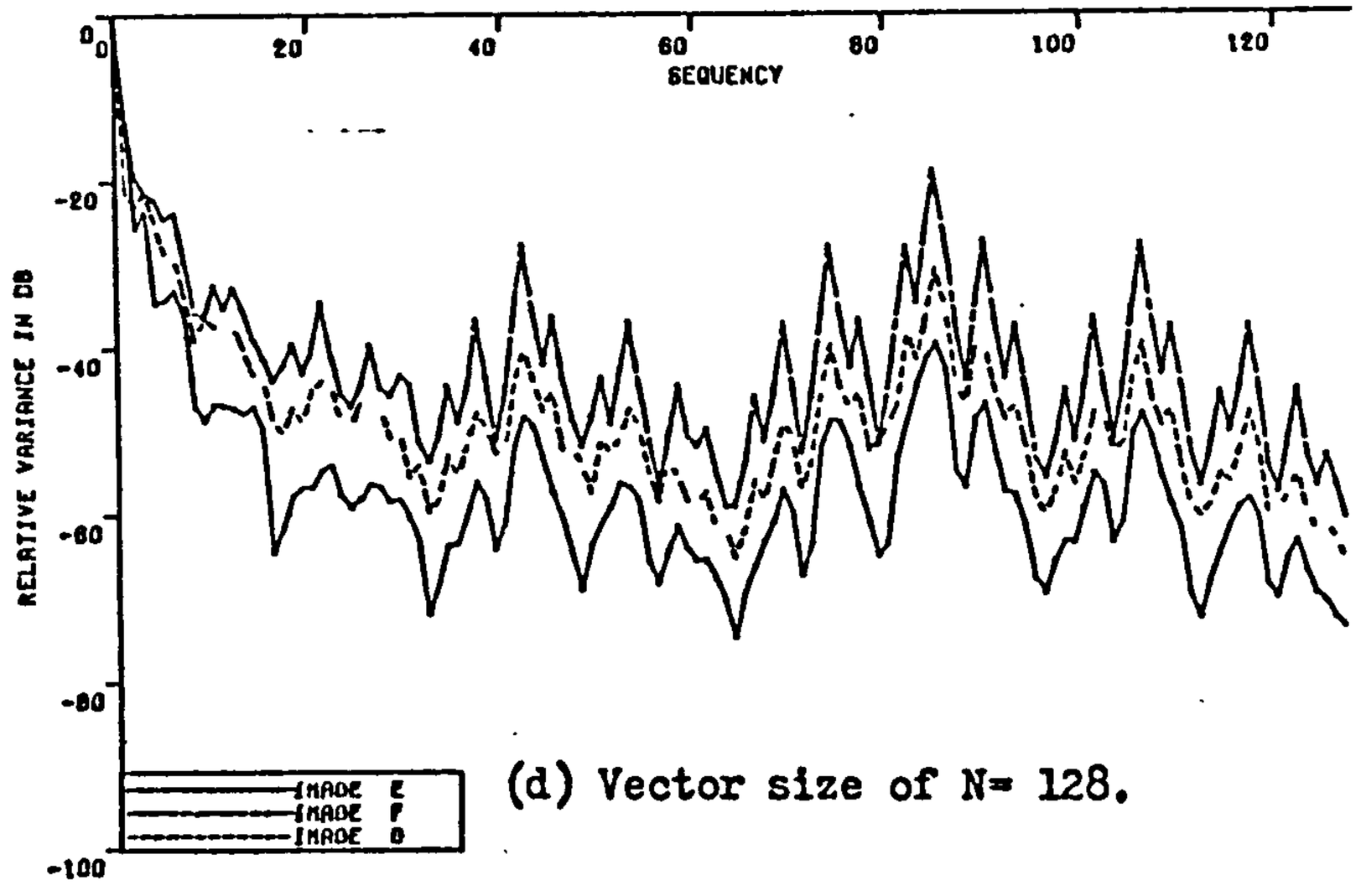
(a) Vector size of N= 16.



(b) Vector size of N= 32.



(c) Vector size of N= 64.



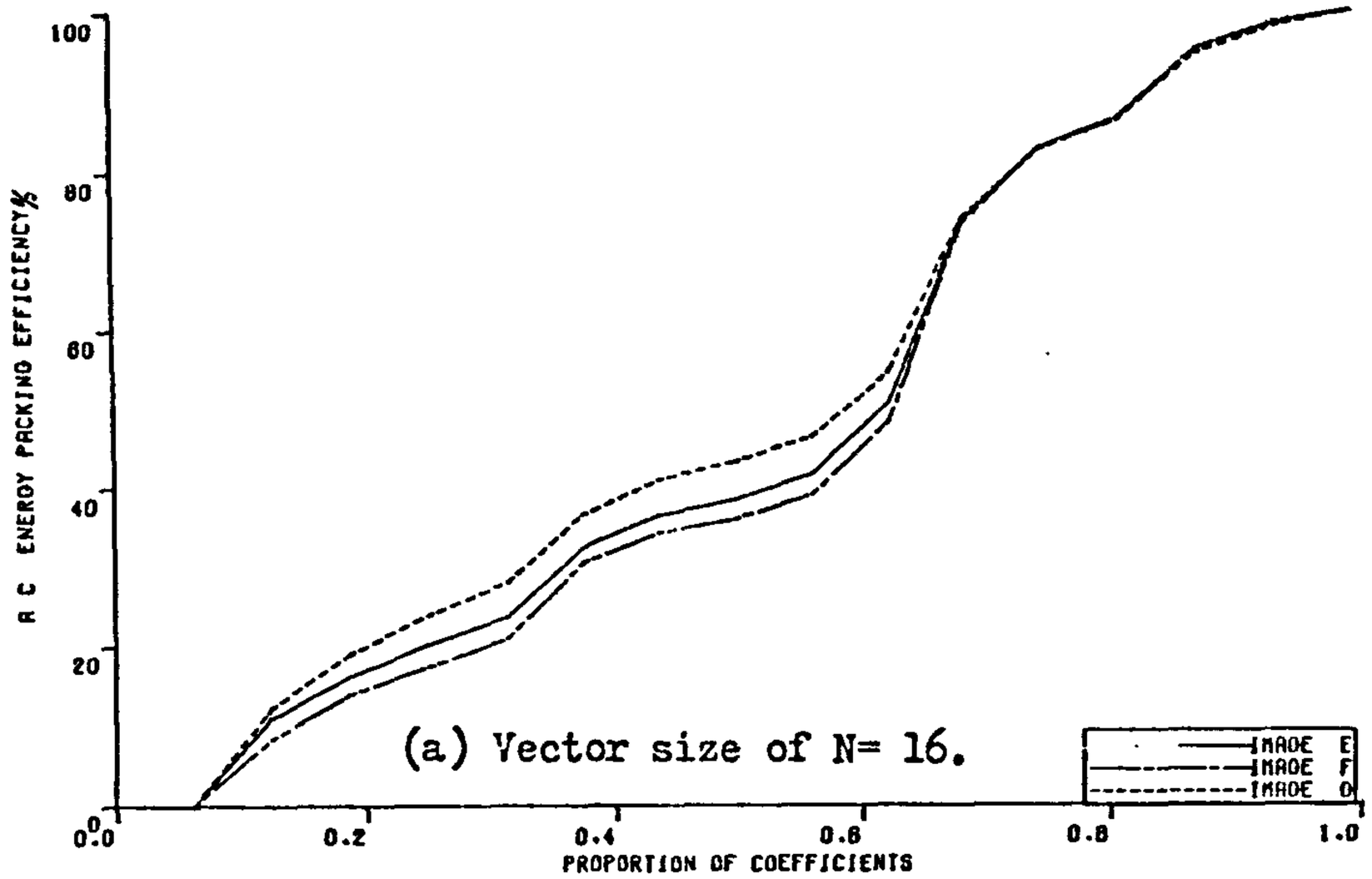
(d) Vector size of N= 128.

Figure (3.3). Relative variances of Direct transform coefficients at $3f_{sc}$.

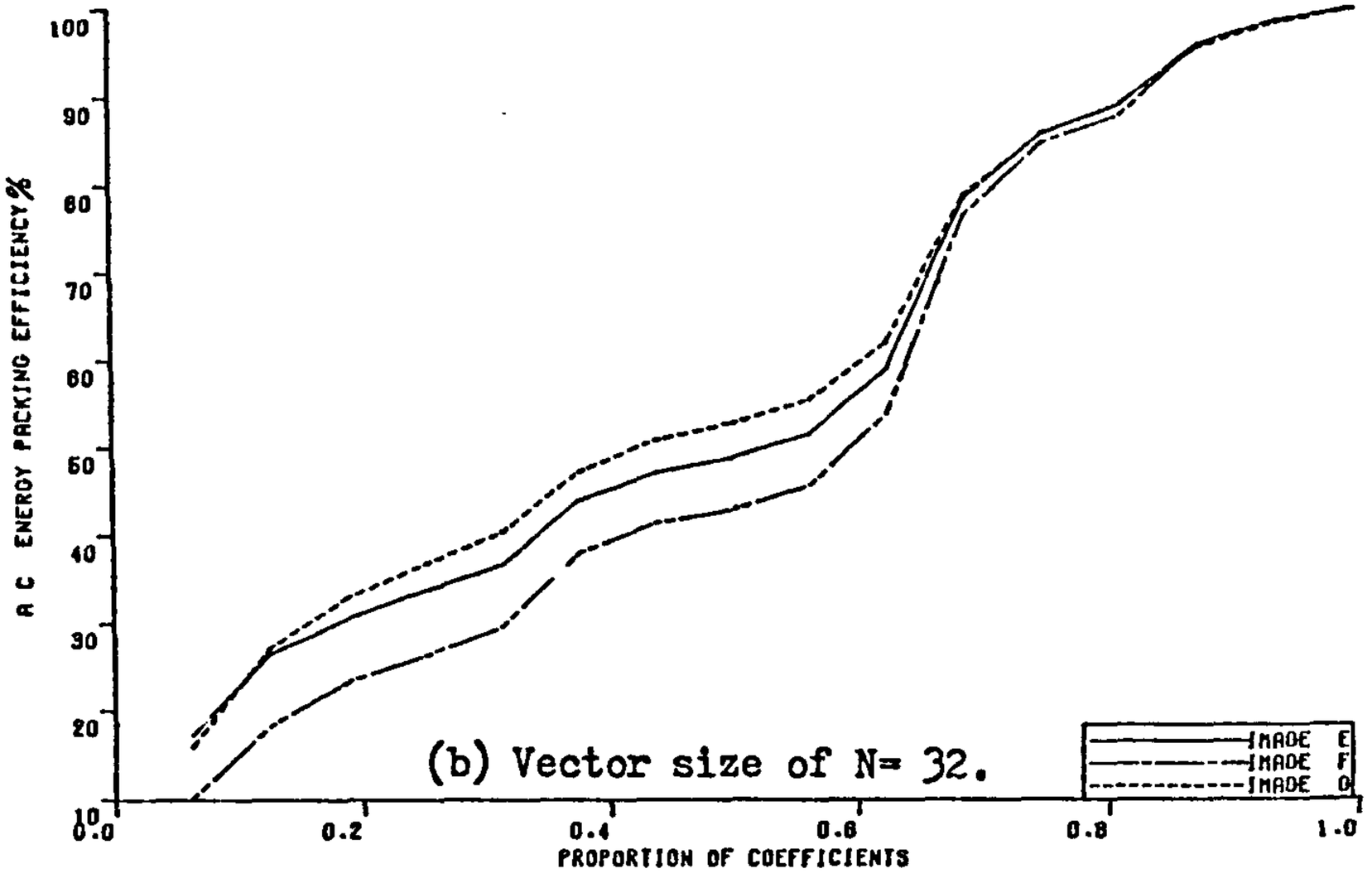
3.1.1.2. Energy Packing Efficiency:

It follows directly from Section (3.1.1.1), that the energy packing efficiency of a direct transform will be poor. Although the energy content in the dc coefficient is still high, (96%- 98%), the ac energy is distributed among more coefficients than in the case of monochrome signals. As already seen in Section (3.1.1.1), some of the medium- and high sequency coefficients have values comparable to, or even higher than, the first ac term H_2. This will result that most of the ac energy will be distributed among these coefficients.

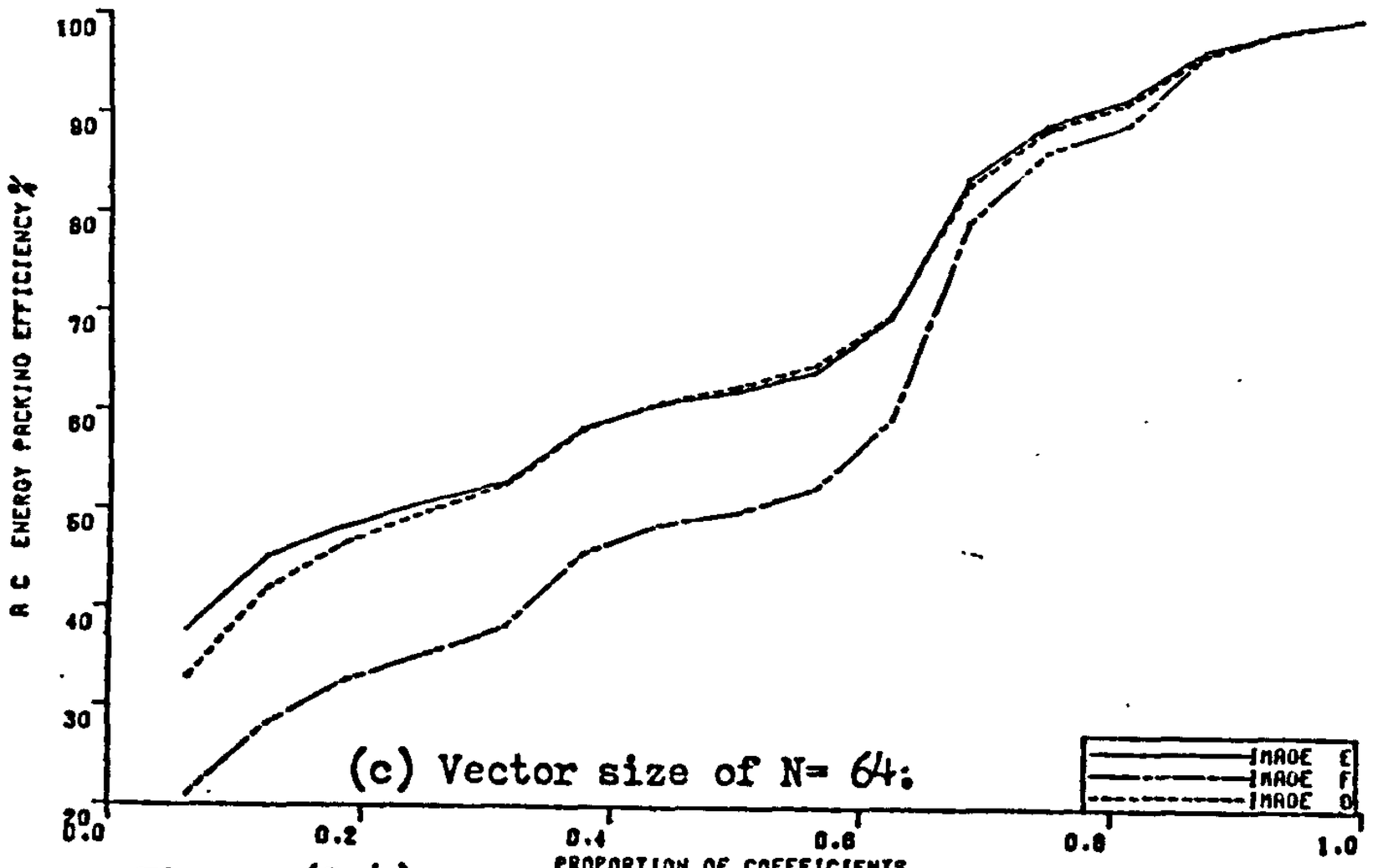
Figure (3.4) shows energy packing efficiency for different vector sizes, and three different image categories. The rise in energy content at proportions of about 0.3 and again at about (0.6- 0.8) is due to the above mentioned distribution. However, these rises are not too sharp to justify any attempt of picking up few coefficients to be coded. As vector size increases, the rises in energy content tend to be more and more slower, thus increasing the proportion of coefficients which must be included to give a satisfactory reproduction.



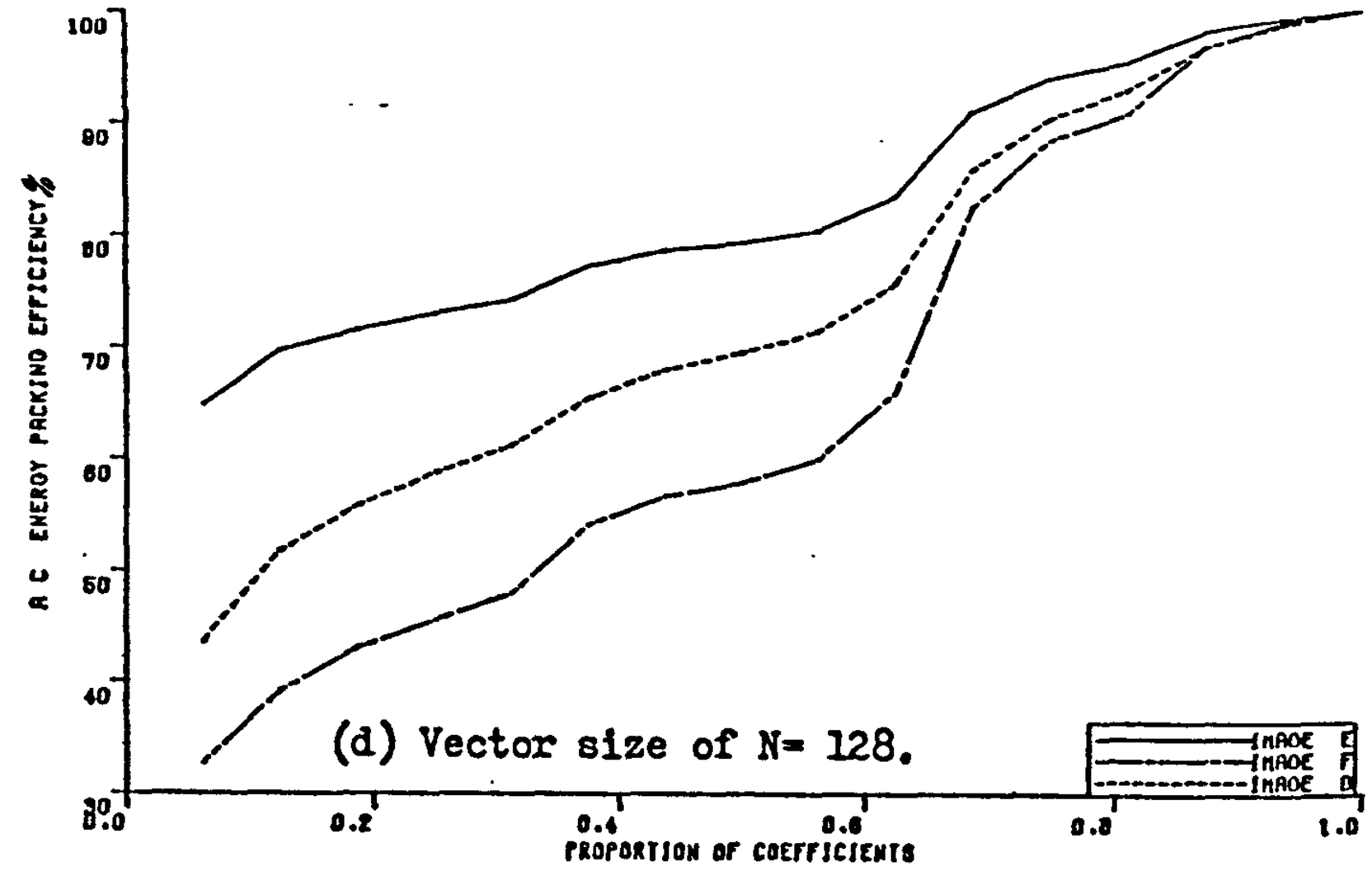
(a) Vector size of N= 16.



(b) Vector size of N= 32.



(c) Vector size of N= 64.



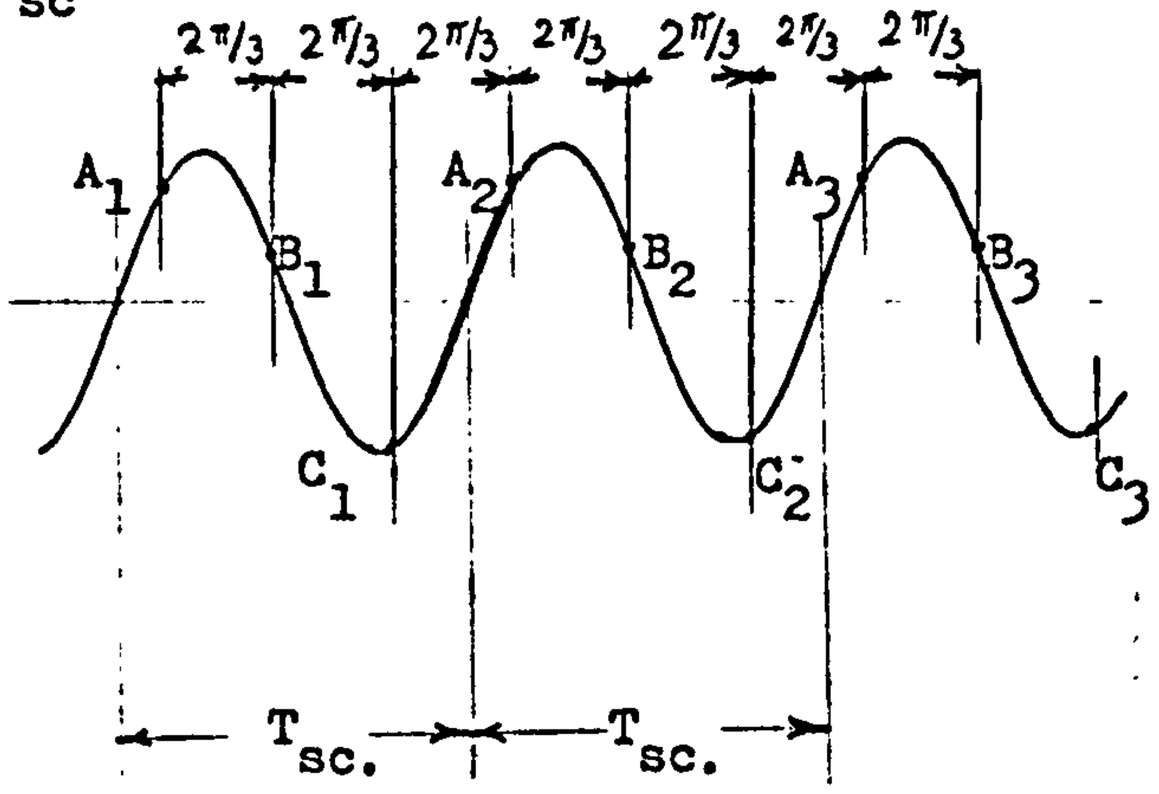
(d) Vector size of N= 128.

Figure (3.4). AC energy packing efficiency for Direct transform at $3f_{sc}$ in different vector sizes.

3.1.2. Laced Transform at $3f_{sc}$:

This type of transform is used in order to overcome the problem of energy distribution among more coefficients in the Direct transform.

Figure (3.5) shows a composite signal of a colour TV line sampled at $3f_{sc}$.



Composite signal.

$T_{sc.}$ = subcarrier interval
= $1/ f_{sc.}$

Figure (3.5). Composite colour signal and samples at $3f_{sc}$.

Starting at any arbitrary point A_1, the two other samples, in a subcarrier interval, are shown as B_1,$2\pi/3$ from A_1, and C_1,$2\pi/3$ from B_1. For the next subcarrier cycle, the samples will be A_2, B_2, and C_2 at similar phases. Due to redundancy in signal, values of A_1& A_2 will differ too little from each other. The same applies to B_1& B_2 , and C_1& C_2 . Extending to several subcarrier cycles is then straightforward, as shown in Figure (3.5). Grouping the A's samples in one group, the B's in another, and the C's in a third, will result three groups, each consists of more correlated samples than the original consecutive ones. Applying Hadamard transform to each of these groups,one would expect results comparable to that of monochrome signals.

For a transform size of N, an N number of subcarrier cycles is taken. The total number of samples will then be 3N. These samples will be considered as forming a so called "Major Vector". Starting with the

first sample, A_1 and 'lacing' each third sample after that will result the first "Minor Vector"

$A_1\ A_2\ A_3\ A_4\ A_5\ A_6\ \ldots\ldots\ldots\ldots\ldots\ldots A_N$,

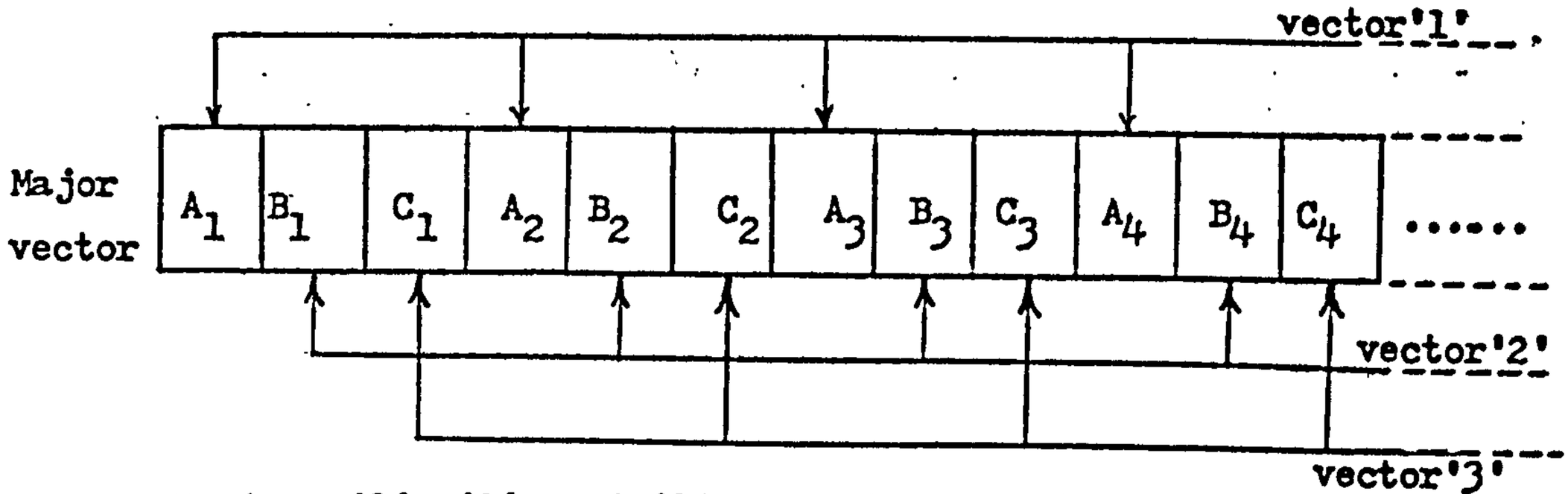
which will be called 'Vector' for simplicity. Repeating, starting this time with B_1, and repeating again starting with C_1,the two other vectors

$B_1\ B_2\ B_3\ B_4\ B_5\ B_6\ \ldots\ldots\ldots\ldots\ldots\ldots B_N$,

$C_1\ C_2\ C_3\ C_4\ C_5\ C_6\ \ldots\ldots\ldots\ldots\ldots\ldots C_N$

are obtained as shown in Figure (3.6).

These three vectors are then processed in the normal way as in the monochrome signals. At the receiving end, the vectors should be 'Delaced' to compose the original 'Major vector'.

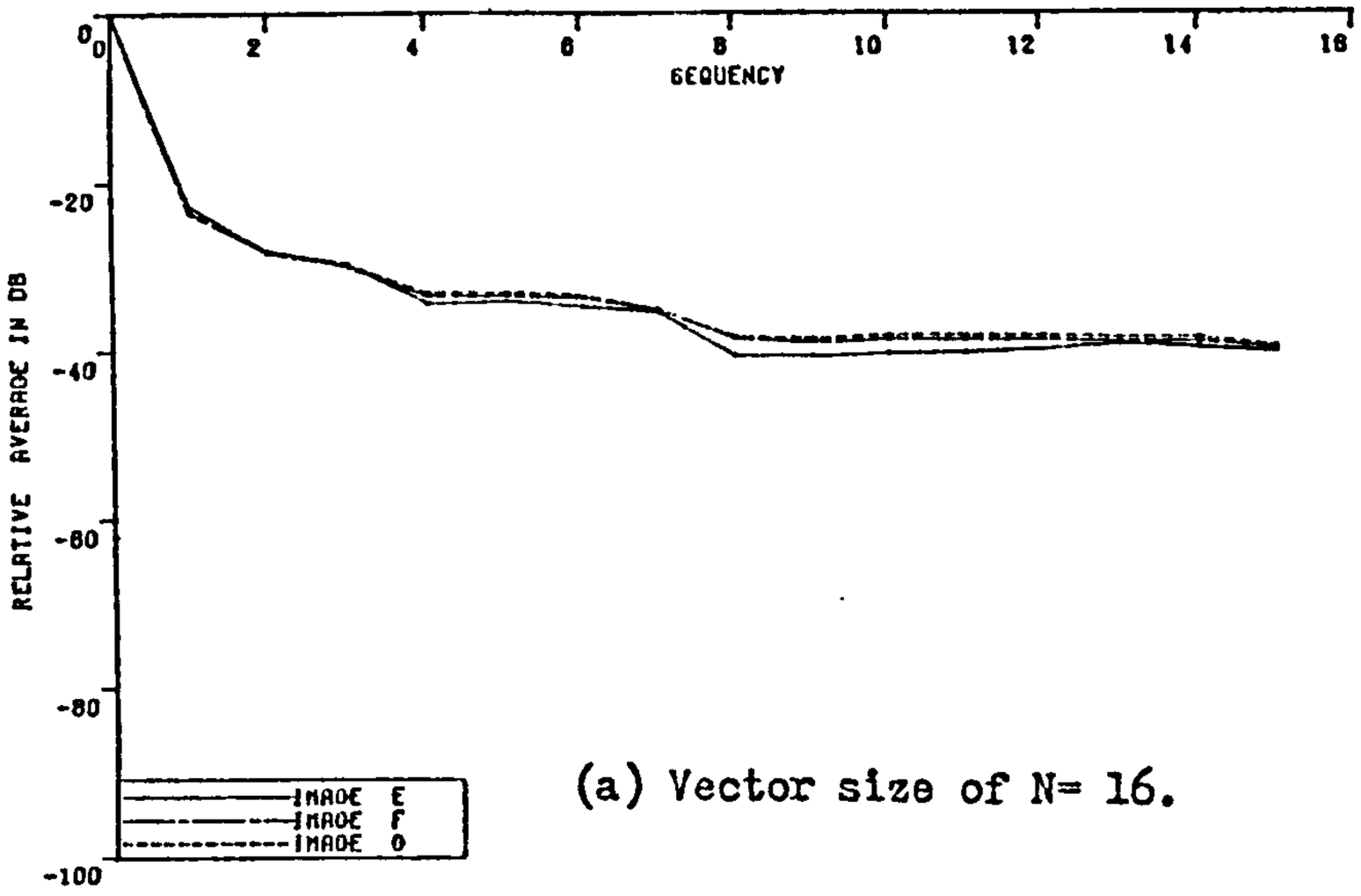


Vectors '1', '2', and '3' are 'minor' vectors.

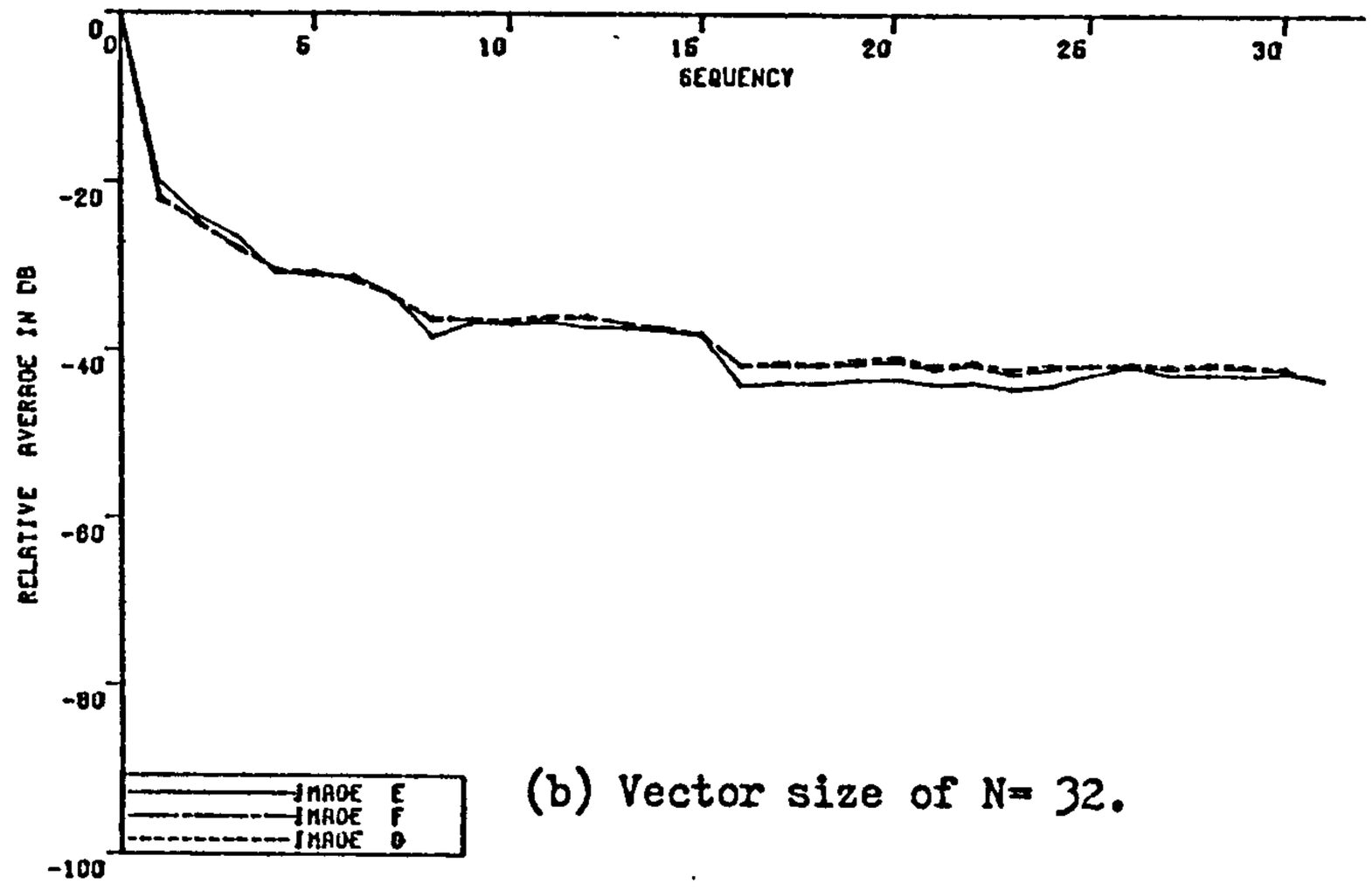
Figure (3.6). Major and Minor vectors in Laced transform.

3.1.2.1. Values and Variances of Coefficients in Laced transform:

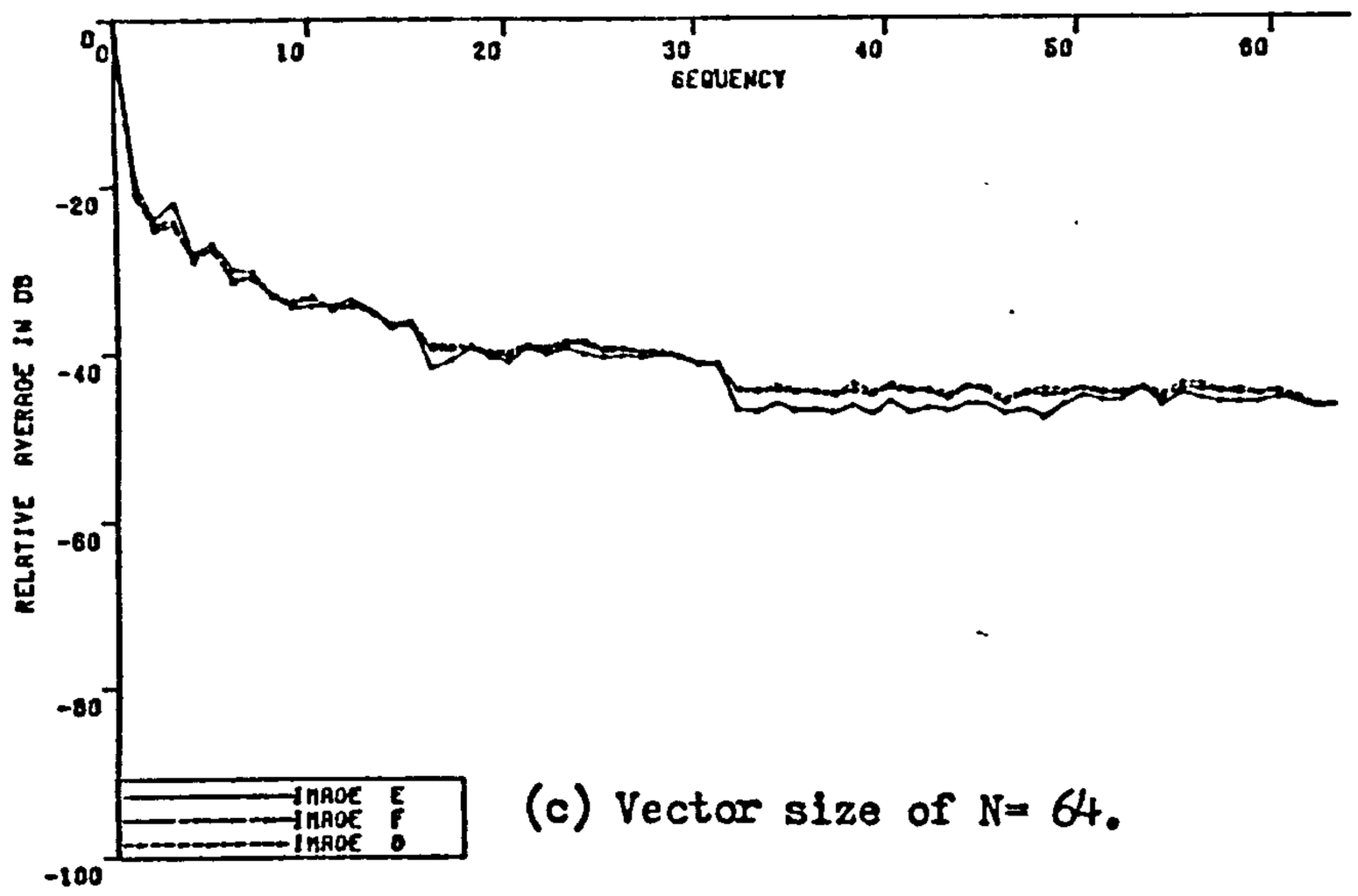
Figure (3.7) shows relative averages for different vector sizes. In general, and as expected from discussions in Section (3.1.2), the behaviour of average values of coefficients in laced transform is nearly the same as in monochrome signals. However, variations among different images are much lower as seems from comparing Figures (3.7) and (2.5).



(a) Vector size of N= 16.



(b) Vector size of N= 32.



(c) Vector size of N= 64.

Figure (3.7). Relative averages of Laced transform coefficients at $3f_{sc}$ for different vector sizes.

A peculiar characteristic of colour signals is still noted, namely the less drop in coefficient values. From Figure (3.7), there is no drop more than 42 dB, compared with 53 dB in monochrome.

Abruptancy in coefficients values at binary boundaries are again noticeable due to high degree of correlation in minor vector samples.

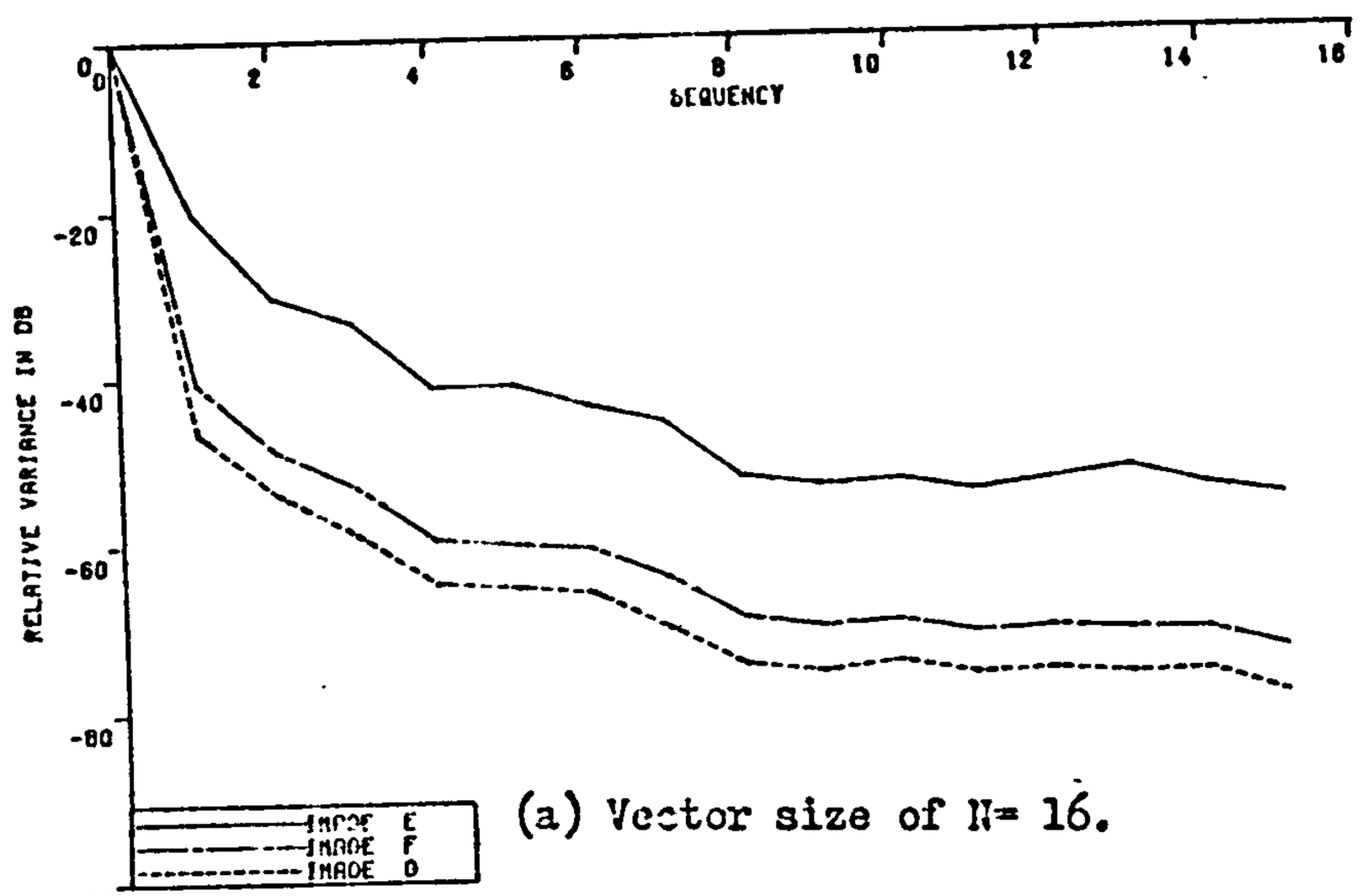
Variances of coefficients are nearly similar to monochrome signals in accordance with Section (3.1.2). Figure (3.8) shows the relative values of variances for some transform sizes.

3.1.2.2. Energy Packing Efficiency:

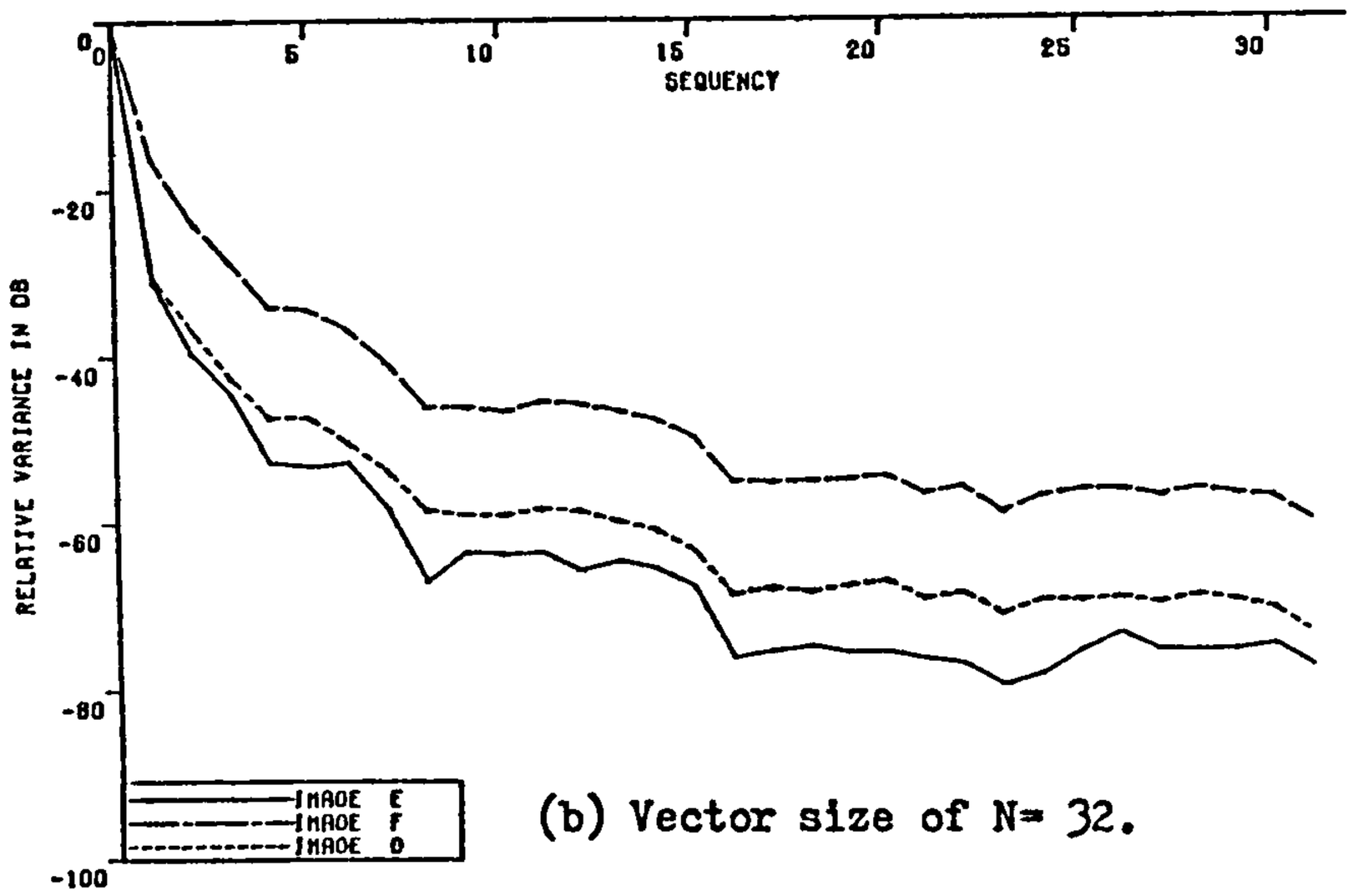
Figure (3.9) shows the energy packing characteristics for different vector sizes and image categories. From these graphs, the packing efficiency is generally comparable with that of monochrome signal transform. As was the case there, the energy content increases considerably with increasing proportion of coefficients in the first half of sequency spectrum.

Variations in packing efficiency with image categories are more noticeable than in cases of coefficients values and variances. The general form of energy packing efficiency function suggests that :

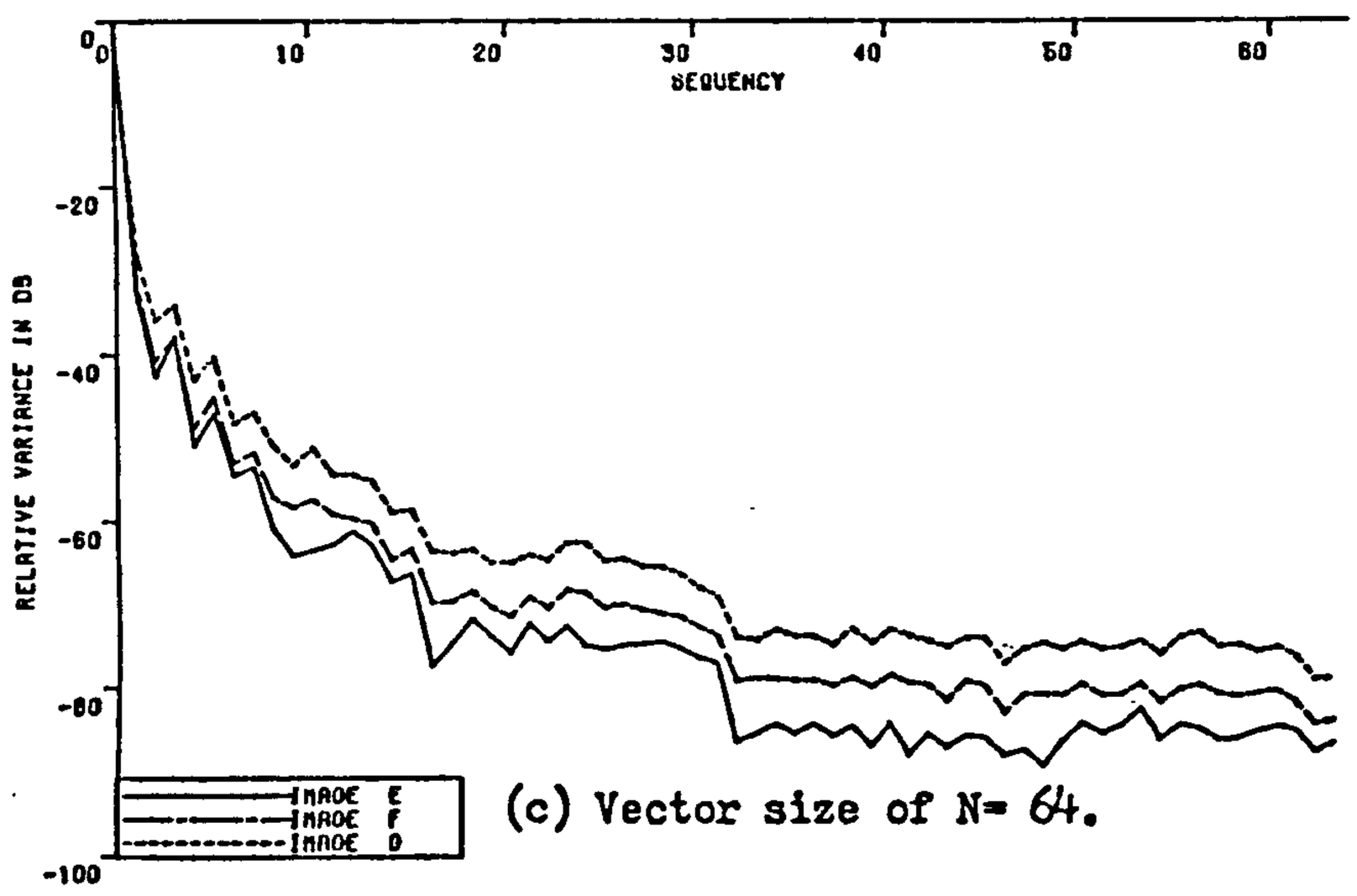
"Laced transform of colour signals could be handled in much the same way as transform of monochrome signals, apart from lacing and delacing".



(a) Vector size of N= 16.

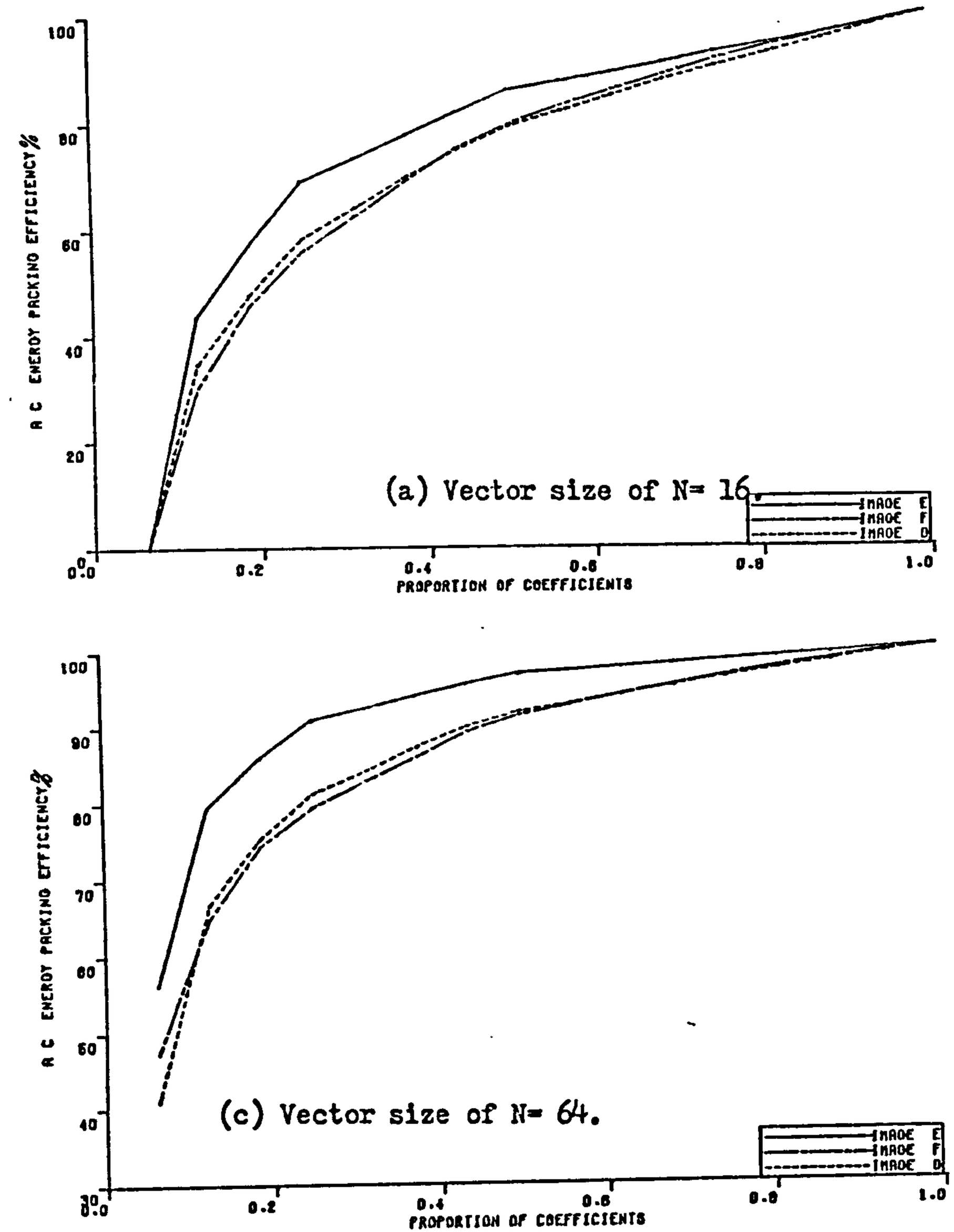


(b) Vector size of N= 32.



(c) Vector size of N= 64.

Figure (3.8). Relative variances of Laced transform coefficients at $3f_{sc}$ for different vector sizes .



(a) Vector size of N= 16.

(b) Vector size of N= 32.

(c) Vector size of N= 64.

Figure (3.9). AC energy packing efficiency in Laced transform at $3f_{sc}$ for different vector sizes.

3.1.3. Components Transform of Colour Signals at $3f_{sc}$:

In this transform, each one of the three components, luminance and two chrominance components, is transformed separately. In fact, this is a case of three transforms, and could be considered as a Plane transformation[65]. Obviously, the total number of transform coefficients will be three times the vector size. Therefore only a substantial reduction in bit rate , if any, will justify its use.

3.1.3.1. Values and Variances of Coefficients:

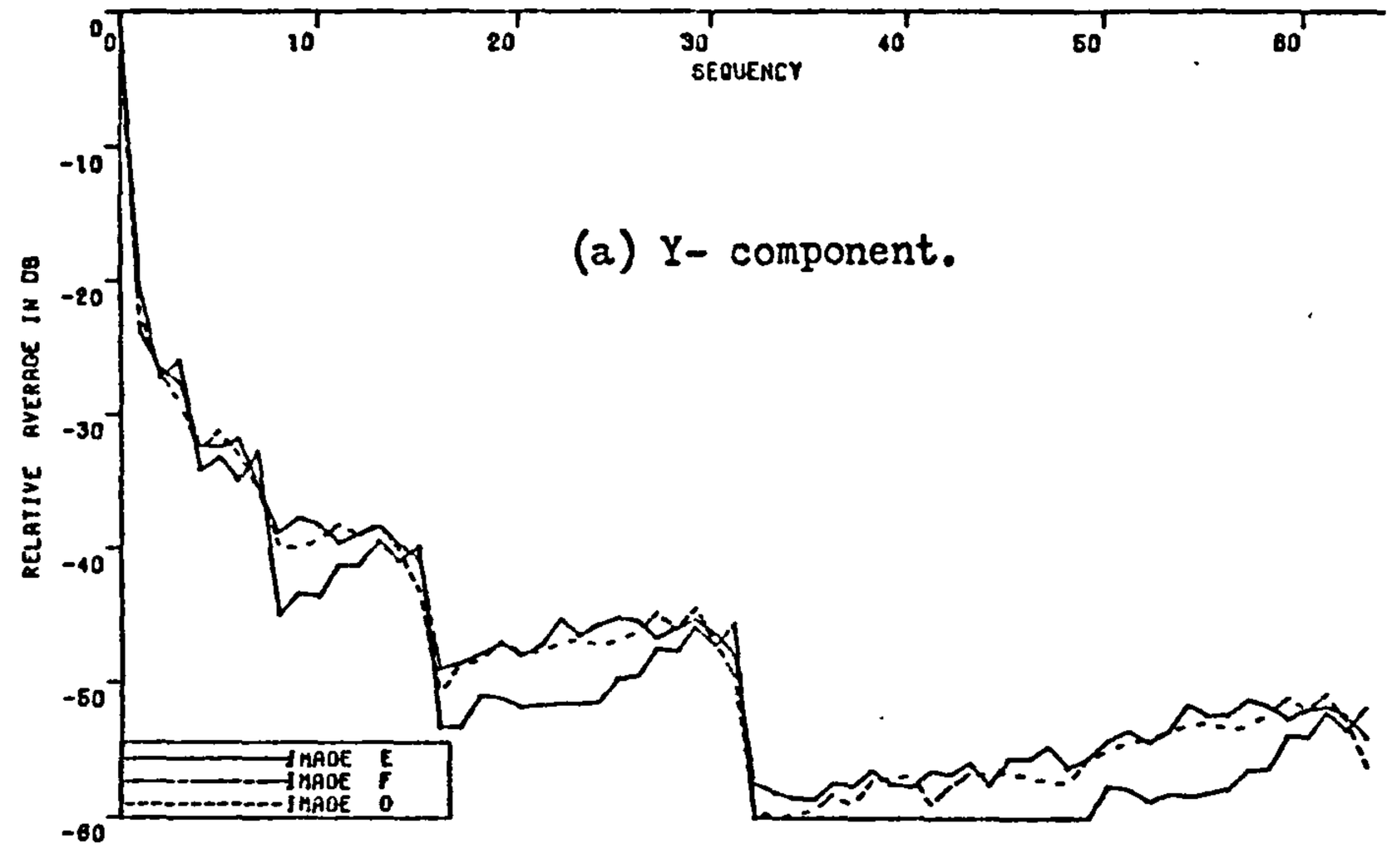
Figure (3.10) shows relative values for an arbitrary vector size of N=64, as an example of components transform.

For the Y (luminance) component averages are generally less than the corresponding values in case of monochrome signals. The decrease in values at binary boundaries are more clear and acute, i.e. abruptancy is sharper. More than three quarters of coefficients are below the -42 dB level. Some variations among image categories are also noted, although they are much lower than in the monochrome case.

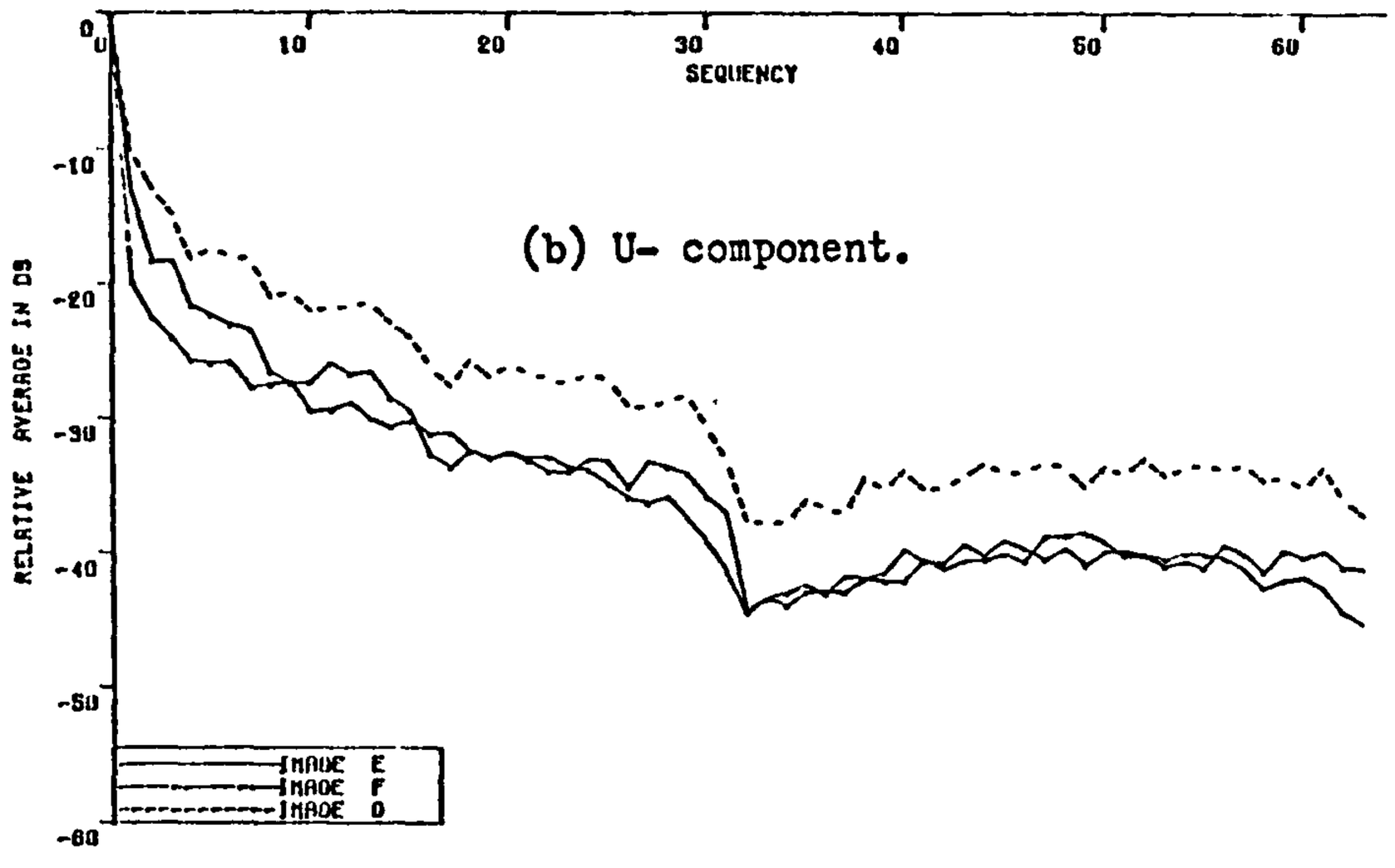
For chrominance components, the two components u and v seem to be nearly similar in characteristics. Abruptancy, though noticeable, is not so clear as in the case of the Y component. This may be due to the fact that, normally, values of u, and v change in spatial domain as colours of particular locations change , while for the Y component, the luminance may be nearly constant for larger areas even though the colours may change. Thus, the degree of correlation in Y component is , in general, much higher than the chrominance components.

There is a general trend that some coefficients in the upper sequency half increase in value again, i.e. the uniformity measure within this sub-vector is not strictly applicable. Variations among different image categories are slightly less than in the Y component, and, consequently, much lower than in the case of monochrome signals.

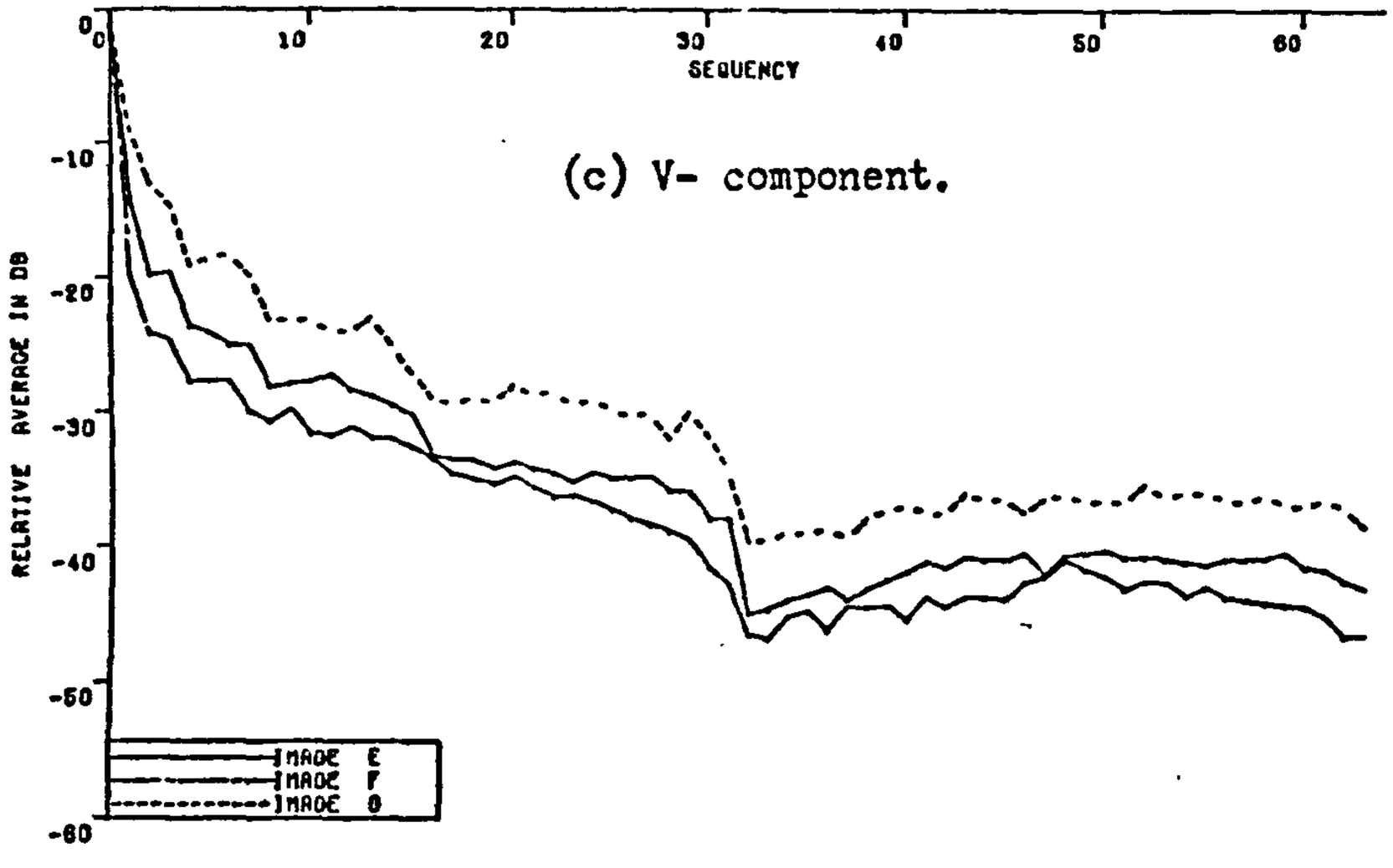
Figure (3.11) shows the relative variances for a vector size N=64.



(a) Y- component.



(b) U- component.



(c) V- component.

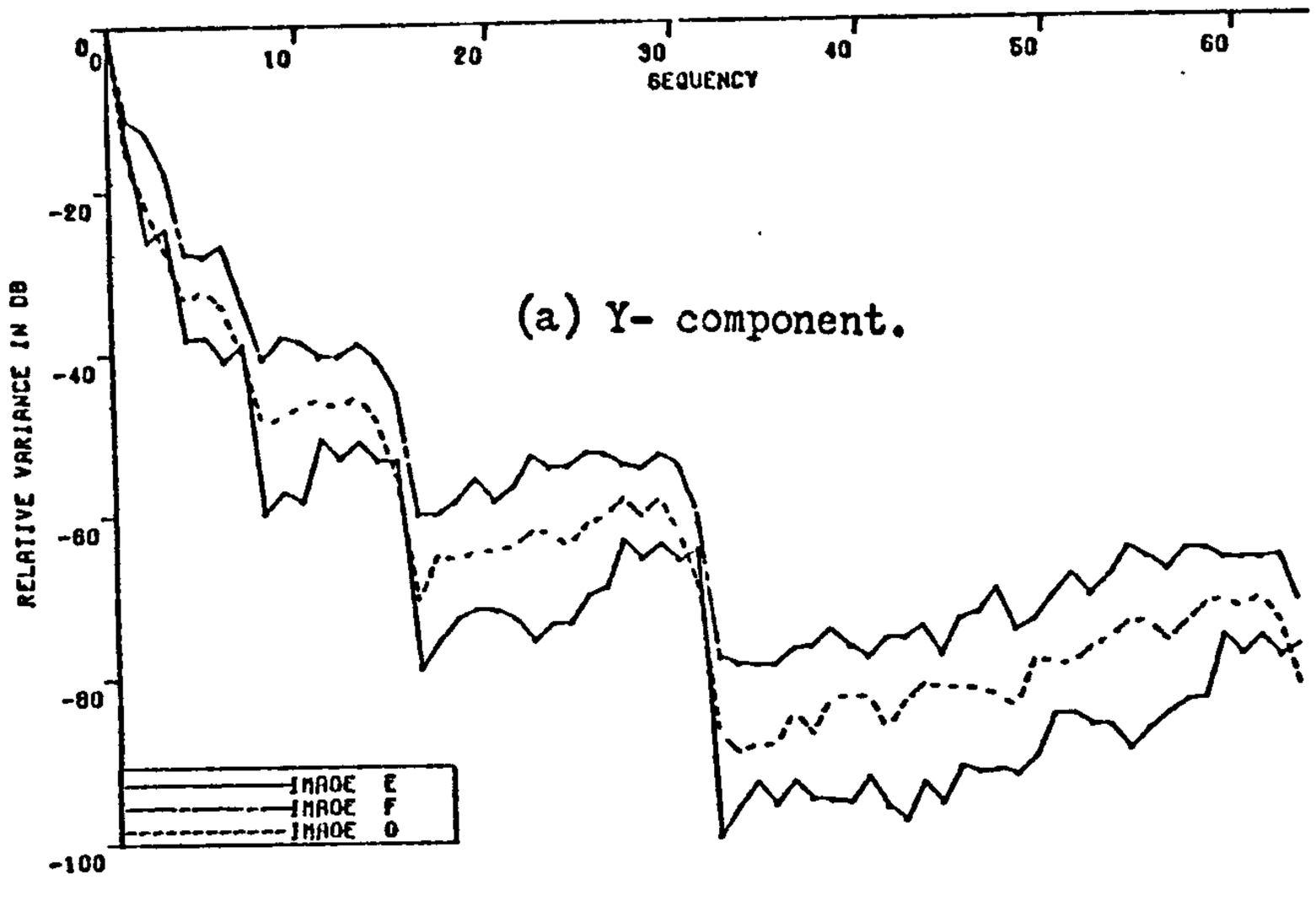
Figure (3.10). Relative averages of coefficient absolute values in Components transform at $3f_{sc}$ for a vector size of N= 64.

3.1.3.2. Energy Packing Efficiency :

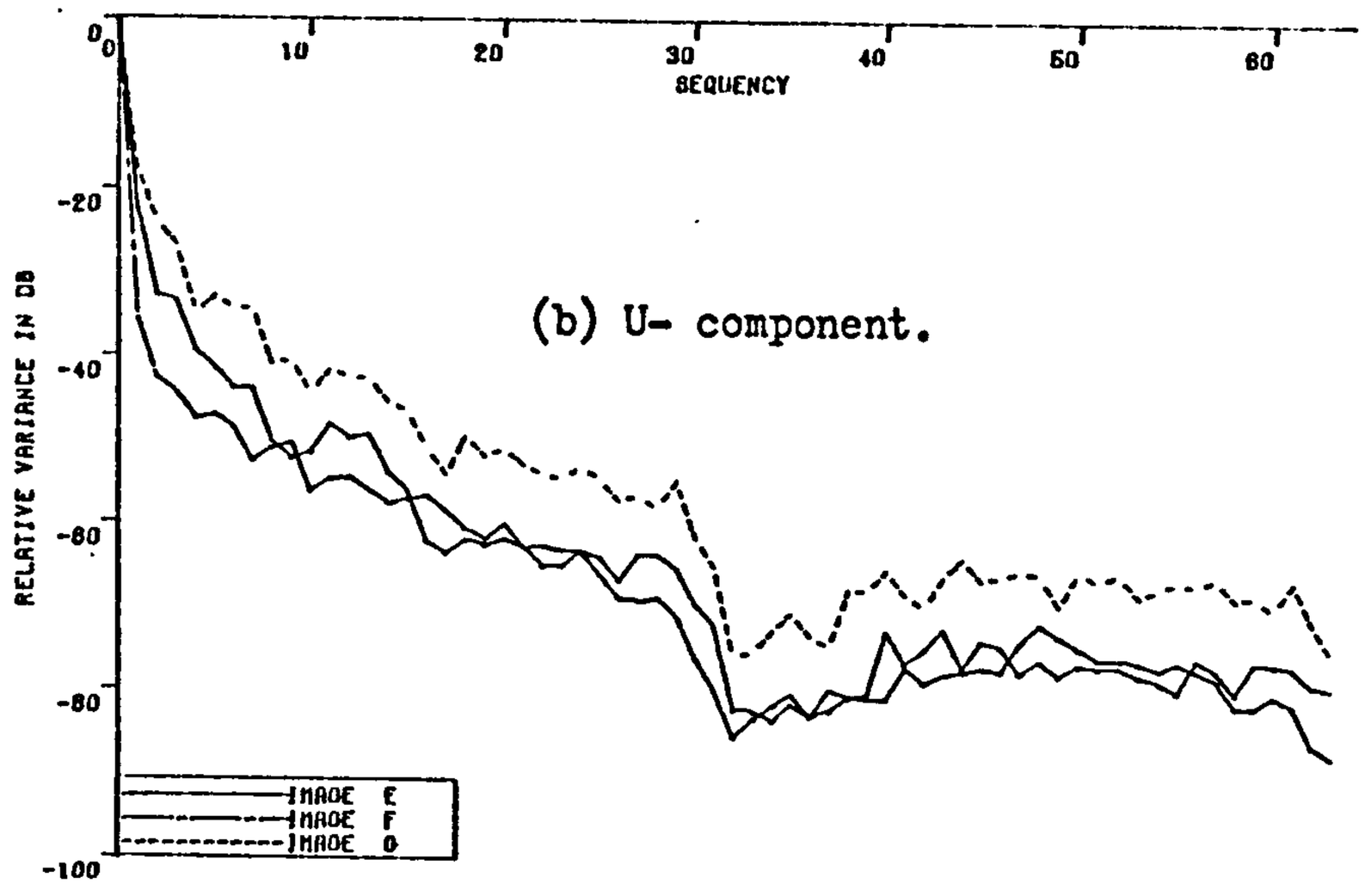
As u and v components may both have negative as well as positive values depending on the colours of particular subpictures, then, unlike the luminance component Y, the first transform coefficient H1 in both u and v transform planes may have any value, small or large, negative or positive alike. Therefore, for the computation of energy packing efficiency in this transform, the dc term in both u and v transform domains will be included. Hence, the energy content for any proportion of coefficients in the u or v transform is shown for all coefficients, including H1.

Figure (3.12) shows the energy packing efficiency for a transform size of N= 64 for the three component transforms in the different images. Figure (3.13) compares the energy packing efficiencies for the three components in each image alone, for the same transform size of N= 64. From the Figures, it can be noticed that the chrominance component transforms u, and v, compact energy more efficiently than the luminance Y component transform. It can be seen also that the variations in packing efficiencies with different images are almost slight.

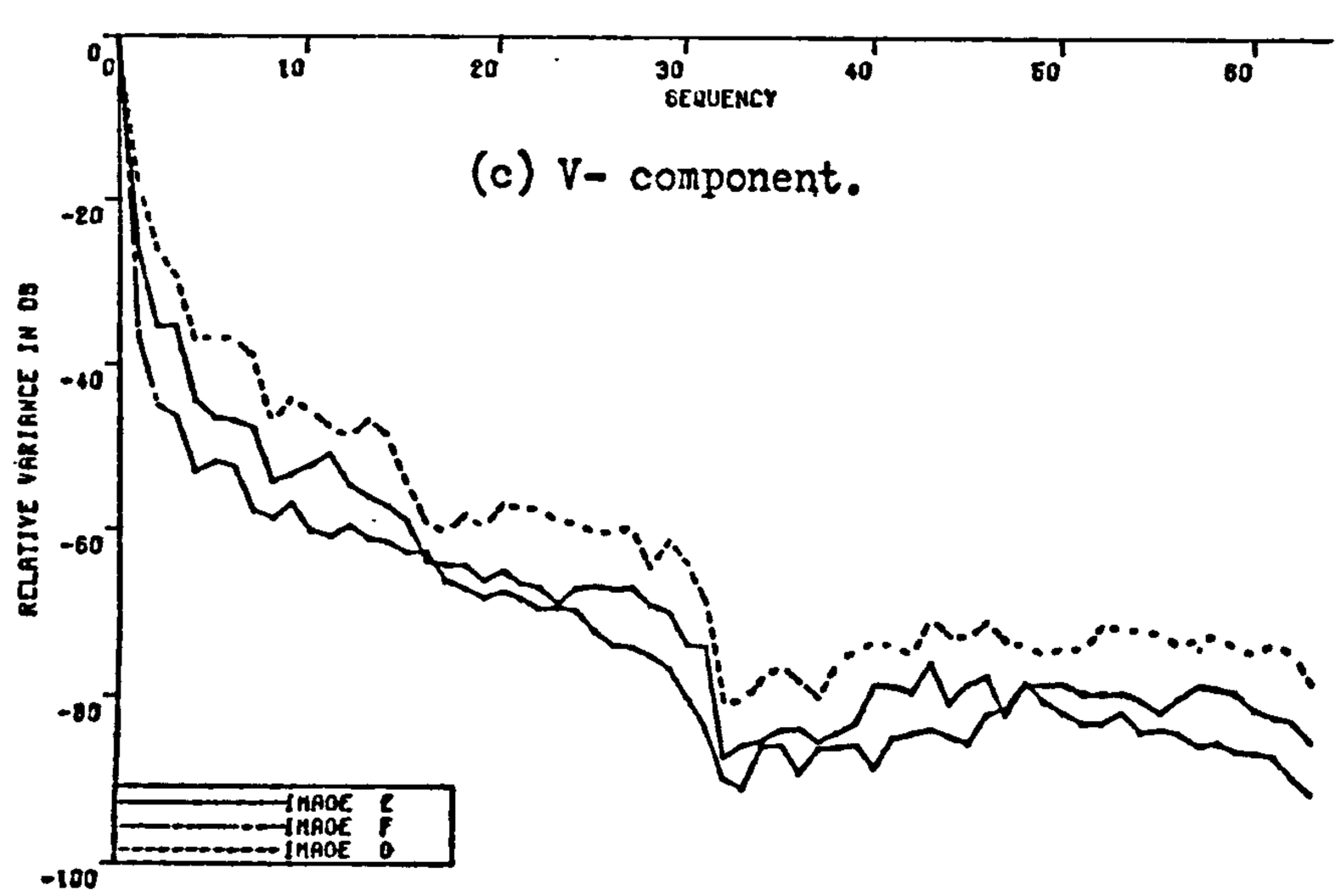
It was found that with increasing the transform size, the superiority of chrominance component transforms of energy packing over the luminance component decreases.



(a) Y- component.

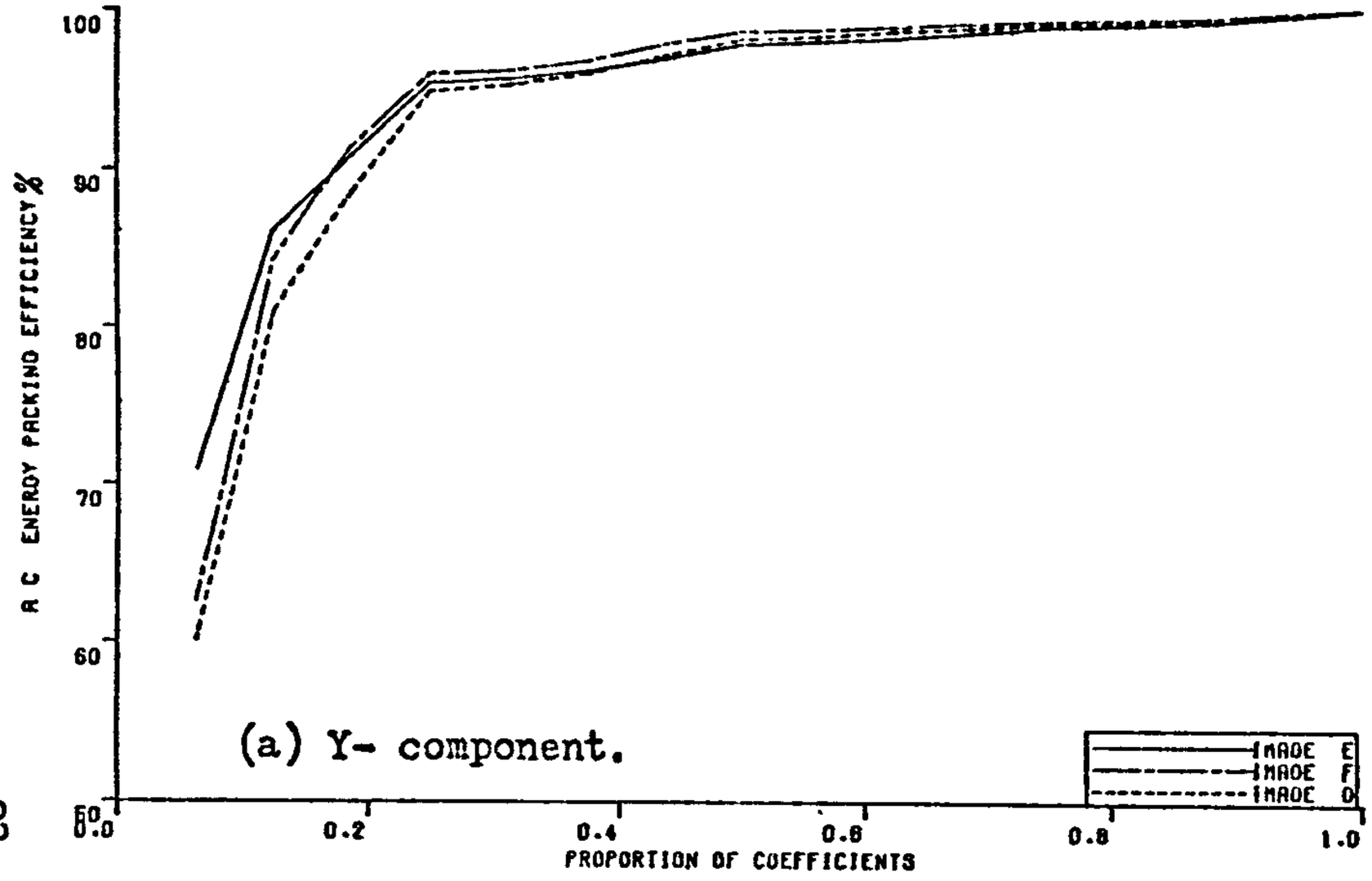


(b) U- component.

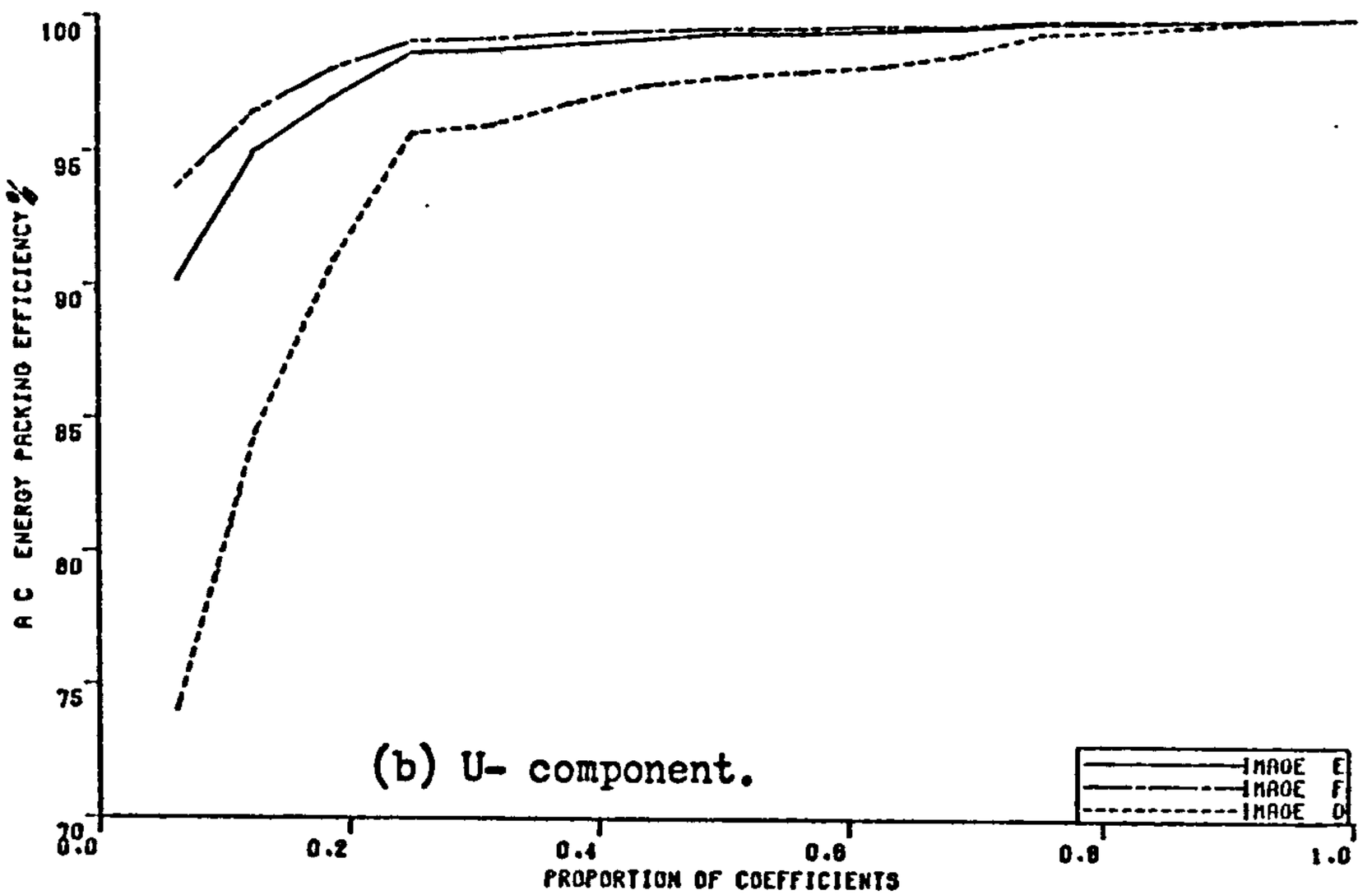


(c) V- component.

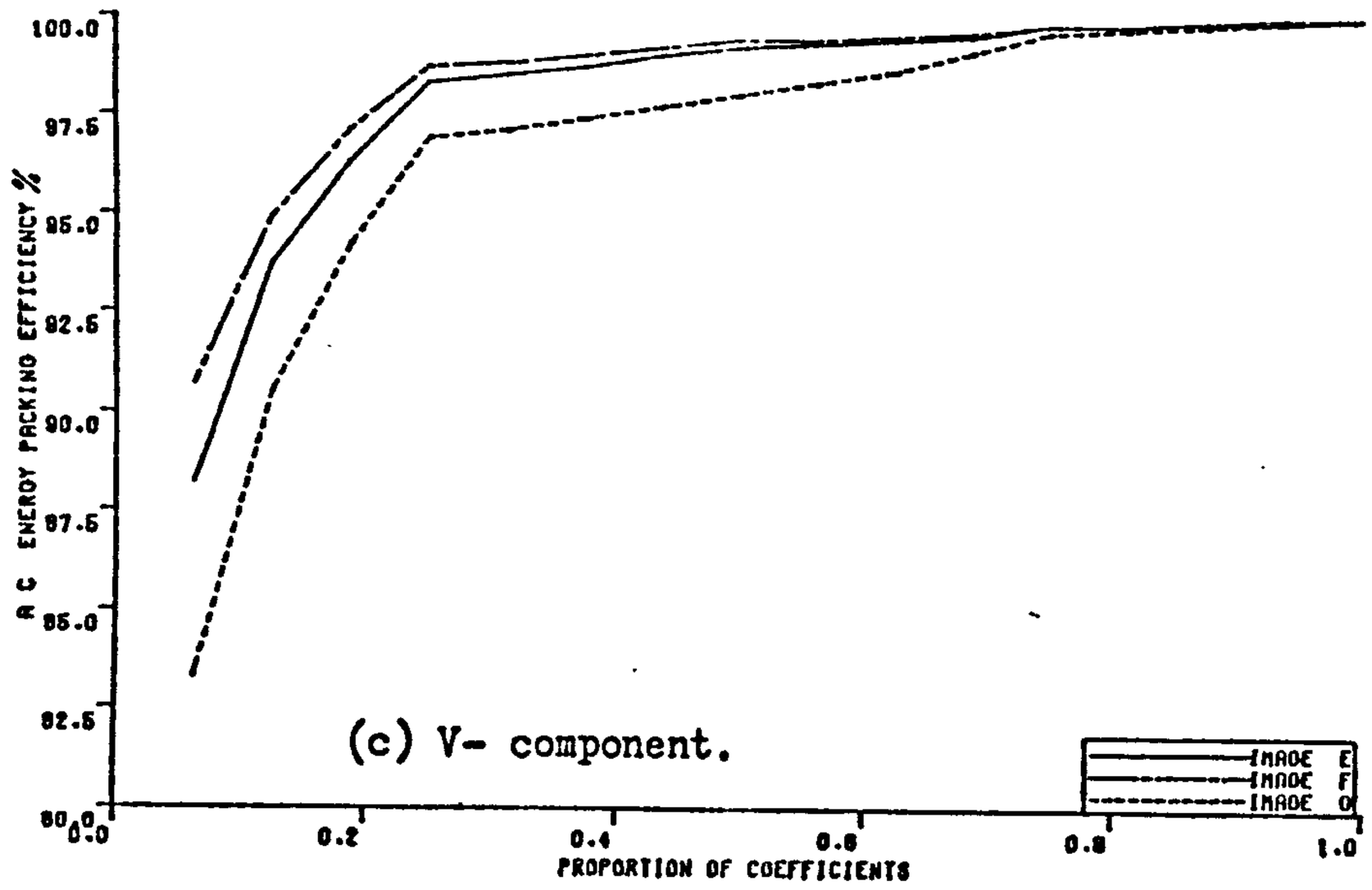
Figure (3.11). Relative variances of coefficients in Components transform at $3f_{sc}$ for a vector size of N= 64.



(a) Y- component.

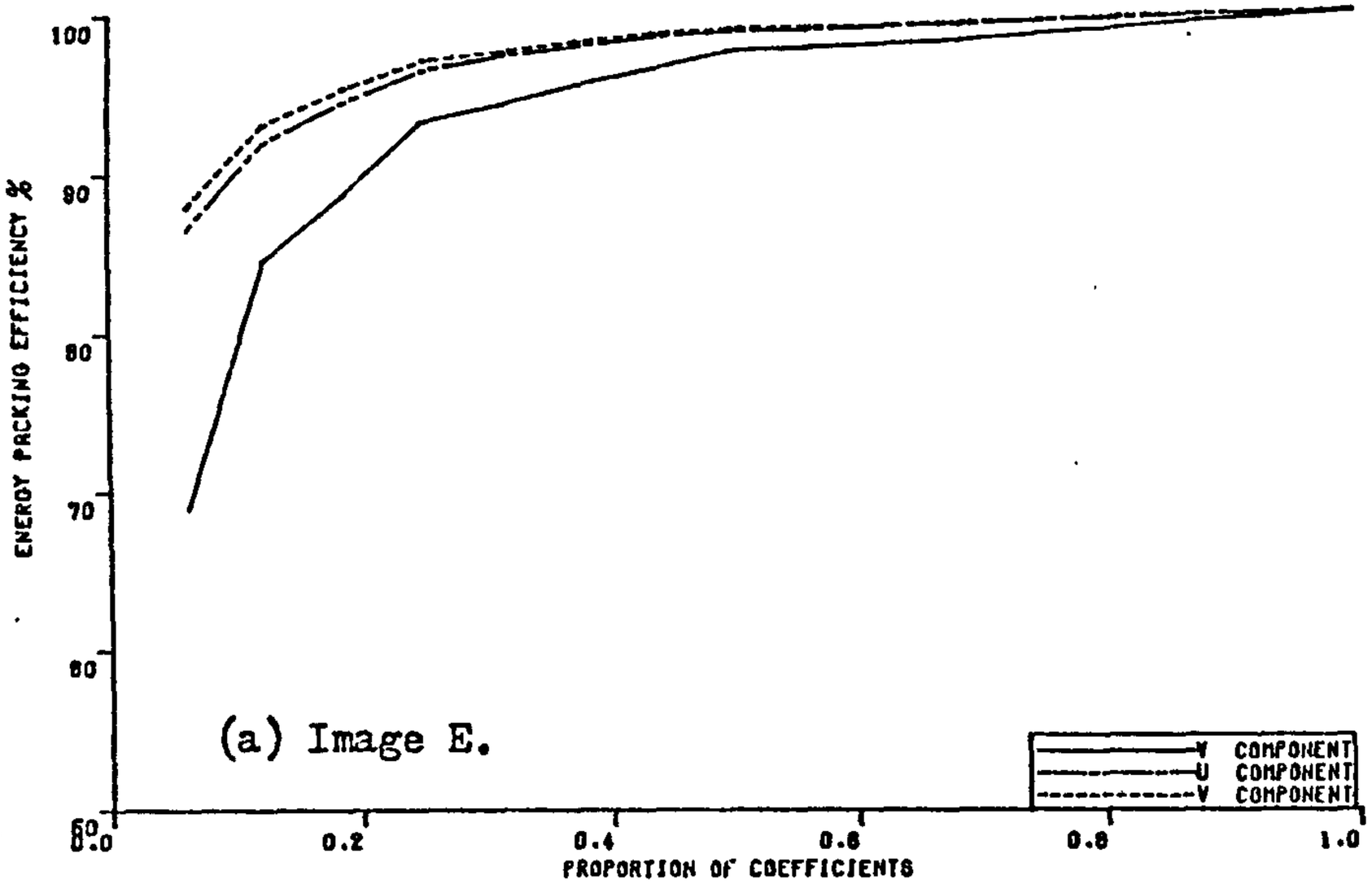


(b) U- component.

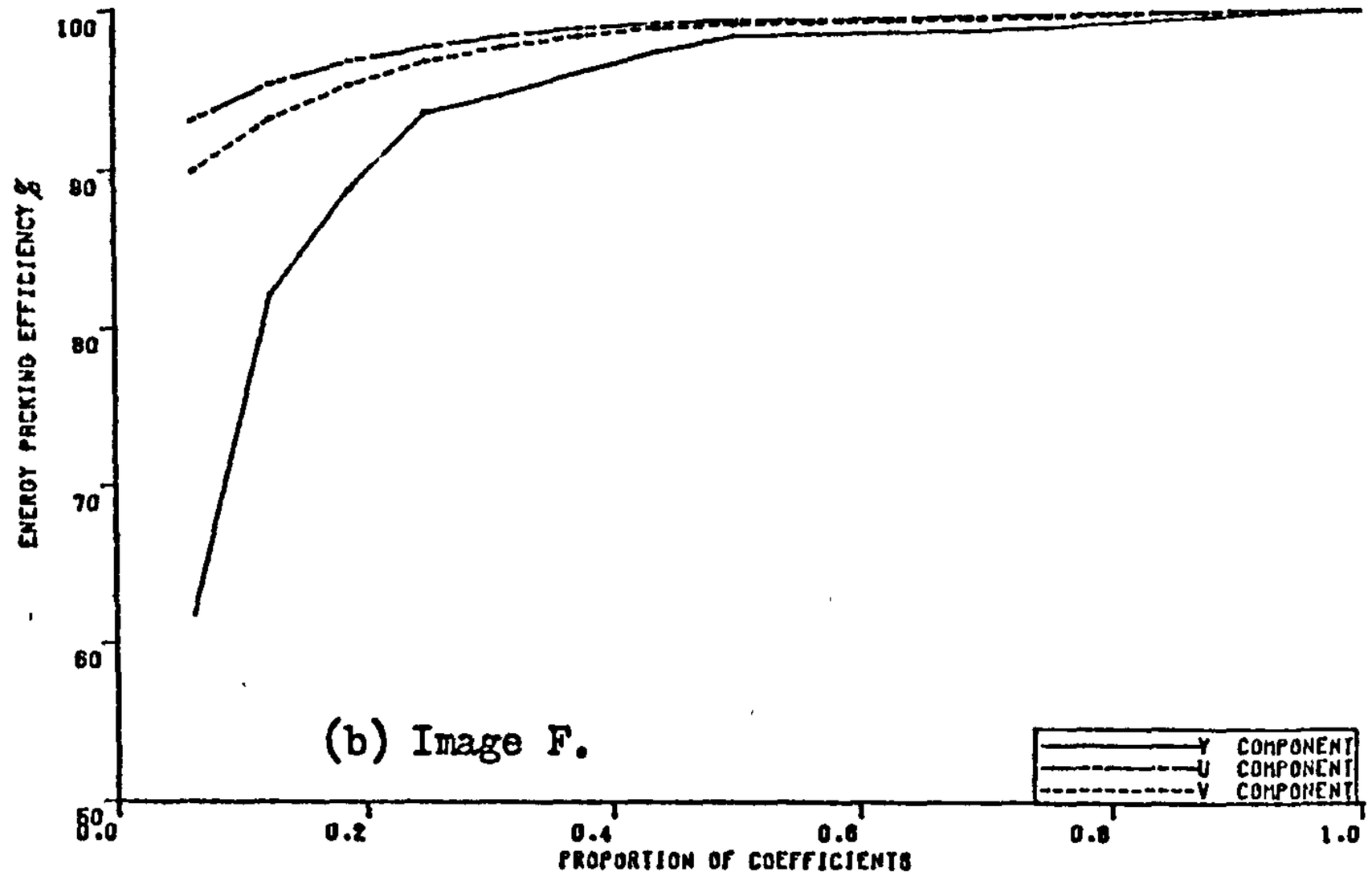


(c) V- component.

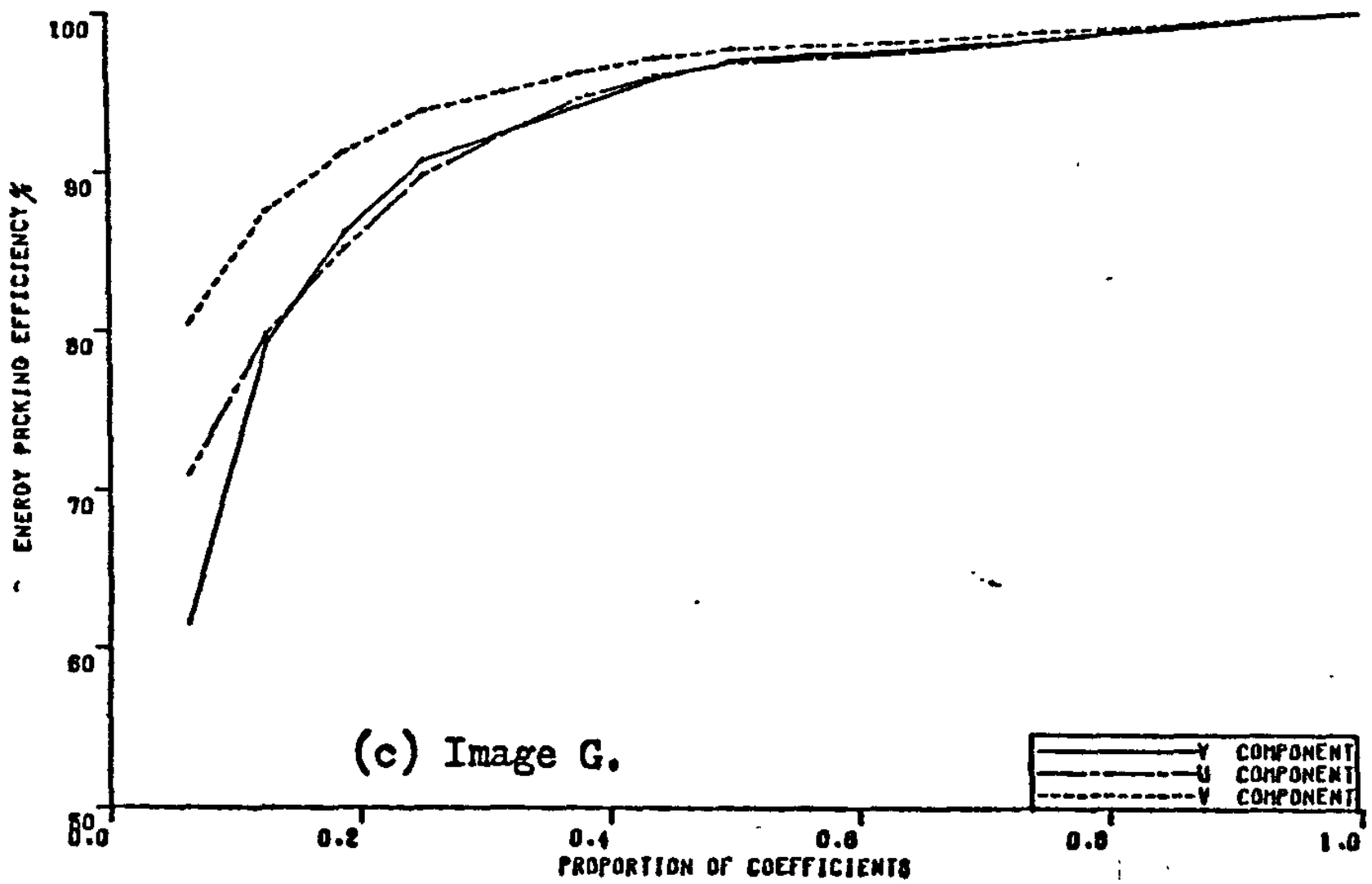
Figure (3.12). Energy packing efficiency in components transform at $3f_{sc}$ for a vector size of N= 64.



(a) Image E.



(b) Image F.



(c) Image G.

Figure (3.13). Energy packing efficiency in components transform at $3f_{sc}$ for different images and a transform size of N= 64.

3.2. Characteristics of Transform Coefficients in case of $4f_{sc}$ Sampling:

The phases of sampling for the case of $4f_{sc}$ can be chosen at $\pi/2$, π, $3\pi/2$, and 2π for the + line. Applying these values to the composite colour signal equation

$$e = E + U \sin wt \pm V \cos wt$$

the four sample values in a subcarrier interval will be

$$A = E + U \quad \text{at } wt = \pi/2 \quad \ldots\ldots (3.4)$$

$$B = E - V \quad \text{at } wt = \pi \quad \ldots\ldots (3.5)$$

$$C = E - U \quad \text{at } wt = 3\pi/2 \quad \ldots\ldots (3.6)$$

$$D = E + V \quad \text{at } wt = 2\pi \quad \ldots\ldots (3.7)$$

Equations (3.4 - 3.7) show that the total sum of all the four samples will be a function of luminance component E only. Also, for addition and subtractions of some sample values, either functions of luminance component alone or zero values will result for some cases. This is most suited to the Hadamard transform, which comprises successive additions and subtractions of sample values. In addition, being an integer power of 2, the number 4 associated with the ratio between sampling frequency and subcarrier frequency, means that for any transform size N, which is itself an integer power of 2, an integer number of subcarrier cycles will be involved, a feature was not met in the case of $3f_{sc}$. These two facts are likely to make the Direct Composite Transform at $4f_{sc}$ sampling more promising.

3.2.1. Direct transform at $4f_{sc}$:

consecutive samples are grouped together to form the transform vector. As seen from Equations (3.4 - 3.7), simple relationships exist among the four samples of a subcarrier interval. Due to redundancy in TV signals, and because a subcarrier period is extremely short in real spatial domain, values of corresponding samples in following interval will be expected to have very close values.

For simplicity, in dealing with sample values, an alternative form of composite signal equation will be used as follows:

$$e = E + P \sin (wt + \phi) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.8)$$

where $p = \sqrt{U^2 + V^2}$

ϕ = arc tan U/V.

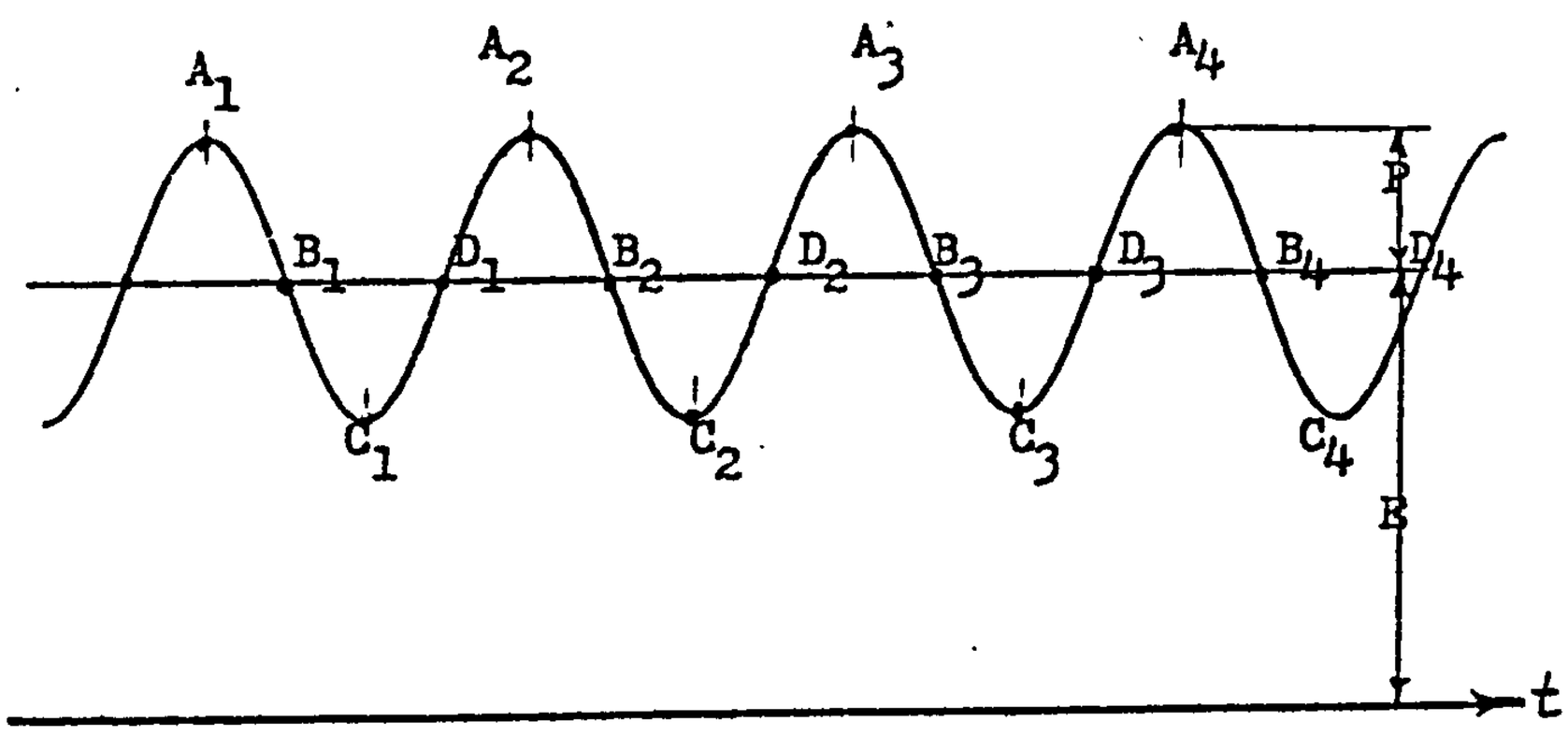


Figure (3.14). Selected sample phases and sample values in a $4f_{sc}$ Direct transformation.

Figure (3.14) shows the composite signal as expressed by Equation (3.8), and shows values of samples in successive four subcarrier intervals.

From that figure, values of samples will be

$$\left.\begin{aligned} A_1 &= E + p \\ B_1 &= E \\ C_1 &= E - p \\ D_1 &= E \end{aligned}\right\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.9)$$

Again, the simple and special relations among the samples values will lead to a high degree of decorrelation in the transform domain.

3.2.1.1. Averages and Variances of Coefficients:

As the redundancy is higher in the case of $4f_{sc}$ sampling frequency, average values of most of transform coefficients are expected to be generally lower than in the case of $3f_{sc}$. Figure (3.15) shows relative averages for different vector sizes. At the first glance, a sharp increase in values of the two mid-sequency coefficients is apparent. Values of these two coefficients are even higher than first ac coefficient, which in previous cases continued to be the biggest ac term.This sharp increase in mid-sequency terms can be explained as follows.

Considering, for simplicity, an 8- sample vector taken from those in Figure (3.14). Values of samples of this will be

$$A_1 \quad B_1 \quad C_1 \quad D_1 \quad A_2 \quad B_2 \quad C_2 \quad D_2$$

As the two subcarrier involved are adjacent, an assumption of equality between each two corresponding samples will not be an unjustified one. Hence, rewriting the vector, assuming $A_1 = A_2 = A$ .....etc., then the values cf,samples will be

$$A \quad B \quad C \quad D \quad A \quad B \quad C \quad D$$

Recalling the values of Hadamard transform coefficients, and substituting with values of samples as functions of E and p (Section 3.2.1.), the Hadamard transform coefficients will be as follows:

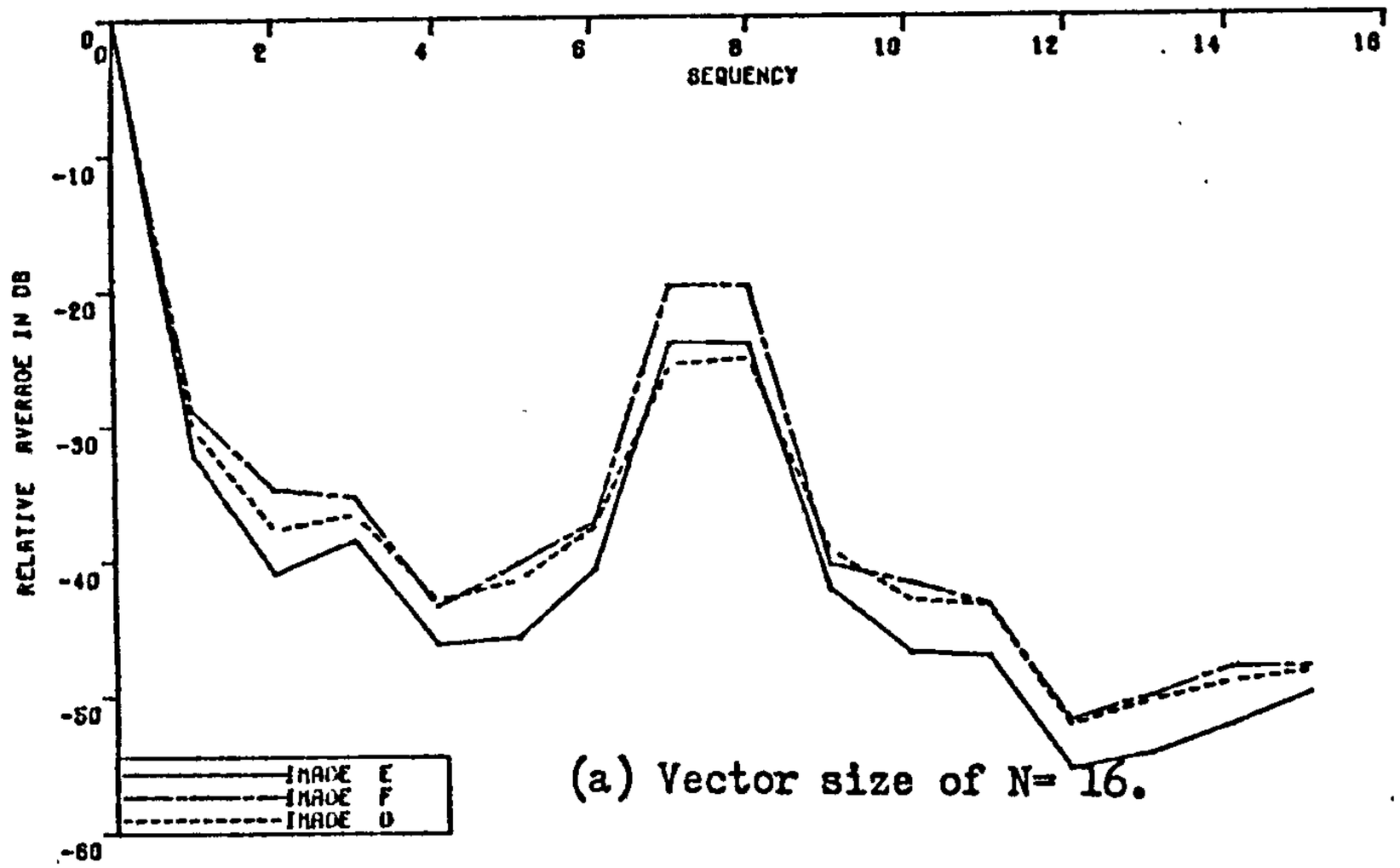
$$
\left.
\begin{aligned}
H_1 &= 8E \\
H_2 &= 0 \\
H_3 &= 0 \\
H_4 &= 4p \\
H_5 &= 4p \\
H_6 &= 0 \\
H_7 &= 0 \\
H_8 &= 0
\end{aligned}
\right\} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3.10).
$$

Thus all of the ac energy is concentrated in the two mid-sequency coefficients H_4 and H_5 , and the dc energy is concentrated in the dc term. Of course, this is an ideal case where there are no changes in values in successive subcarrier intervals. Practically, these changes are so small that the fact of concentration of most of the ac energy in these two coefficients will still exist. The actual values of other coefficients will be very small instead of theoretical zeros in Equations (3.10). Following exactly the same procedures, it could be shown that for bigger transform sizes, as N=16, 32, 64,...etc., the same results will be generally obtained. However, for bigger sizes of transform, the number of differences involved in obtaining transform coefficients will increase, leading to some increases in coefficients which would, theoretically, have zero values and hence, decreasing the values of the two mid-sequency terms.

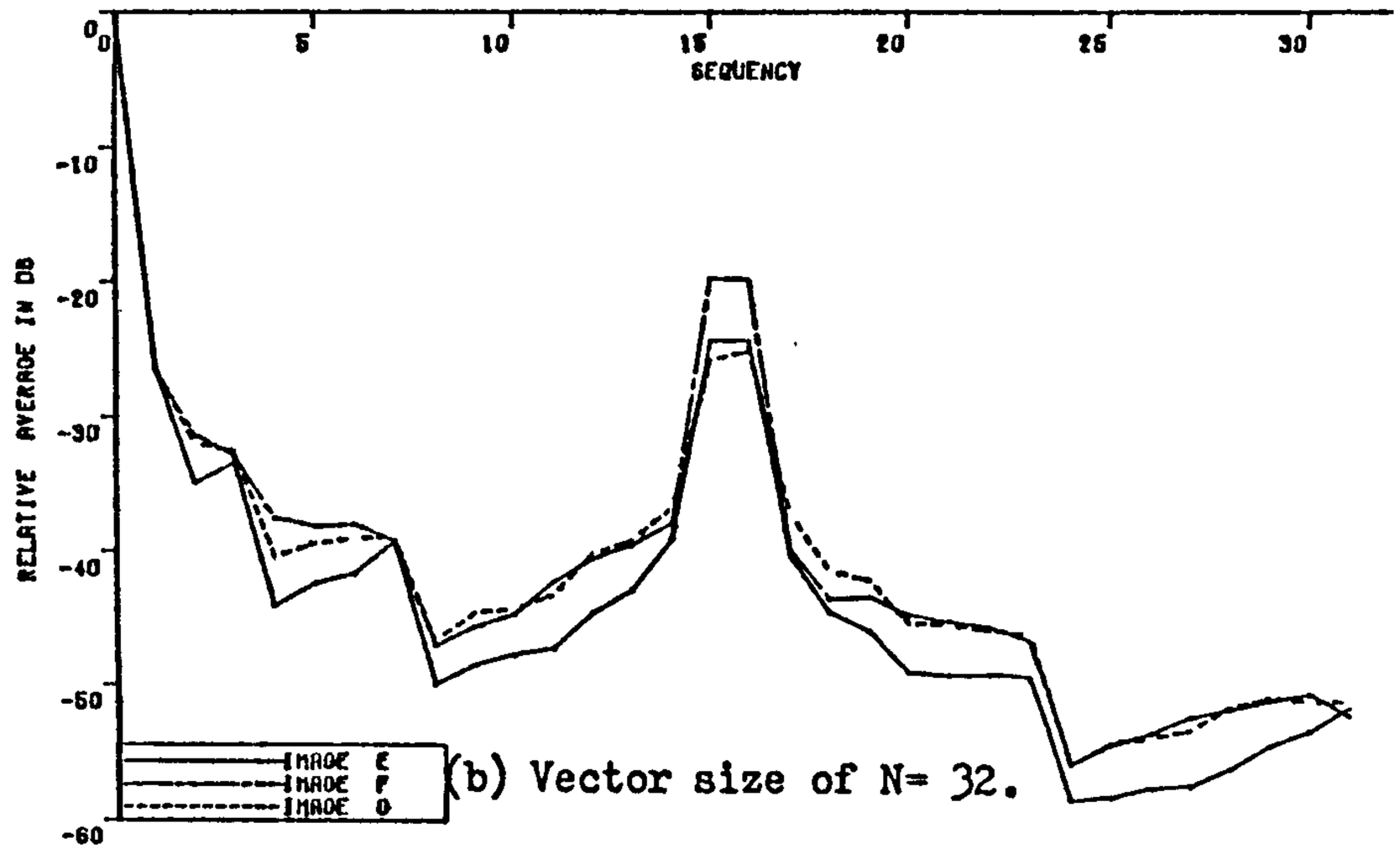
In accordance with this analysis, Figure (3.15) shows the increase in values of the two mid-sequency coefficients and the decrease in other ac terms. Tables (3.4 - 3.6) show the proportions of coefficients having definite levels relative to dc term. From these tables 70% of terms have values less than -42 dB relative to dc term. The tables also show the increase in relative values of some other coefficients on both sides of mid-sequency terms as the transform size increases, and hence, more and more ac energy is switched to these coefficients. It may be concluded that the extremely good concentration of ac energy in the case of $4f_{sc}$ small

vector sizes suggests that in such a transform three coefficients, the two mid-sequency and the dc one, can play the major part in recovering the original spatial data.

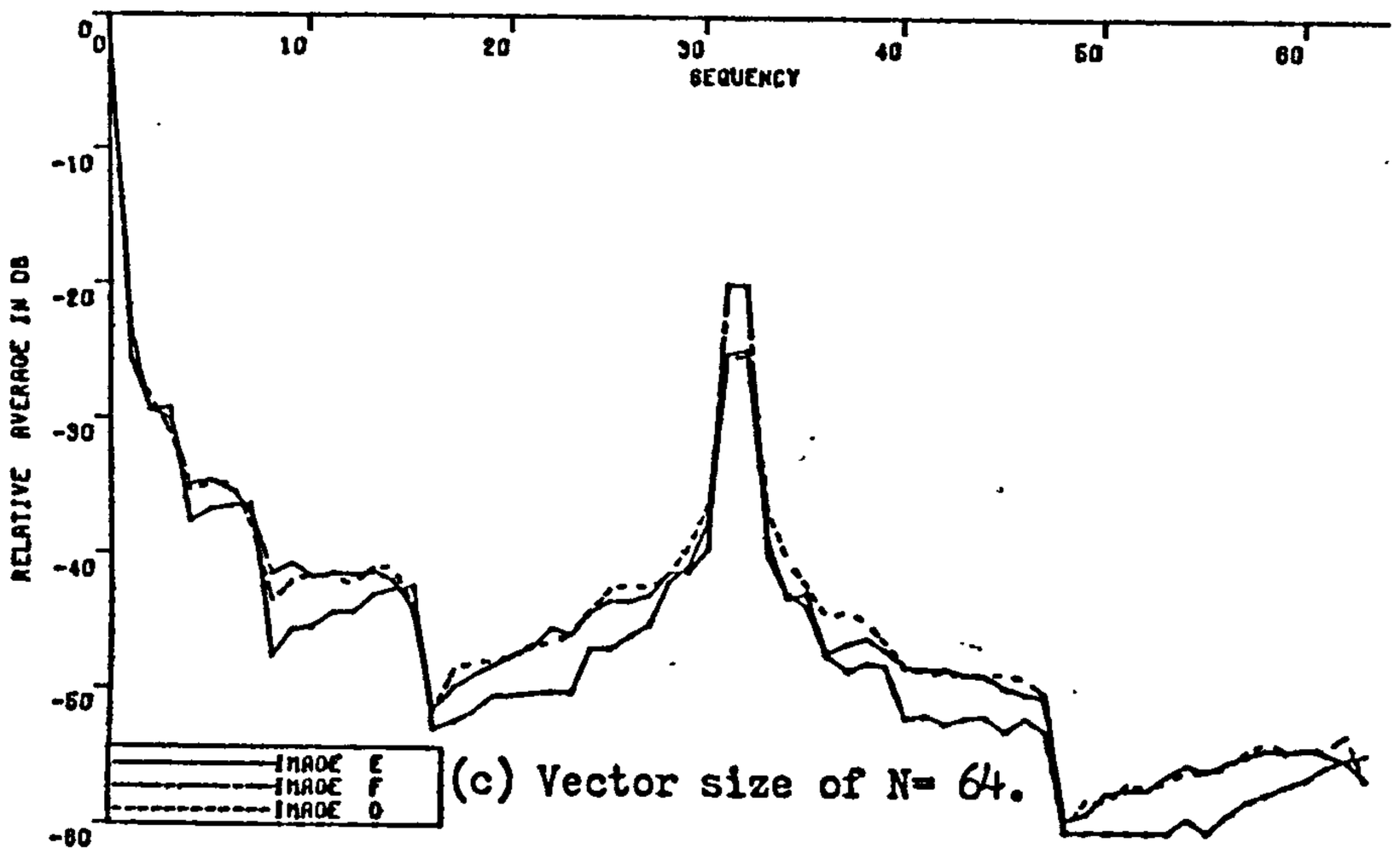
Figure (3.16) shows relative variances for different transform sizes and image categories. It is again apparent that the variances of different coefficients follow nearly the same general characteristics of the average values. Here also the variances for the two mid-sequency terms are the largest among all ac coefficients. Variations among image categories are much more than in the case of averages. As the transform size increases variances of more and more coefficients on both sides of mid-sequency terms also increase.



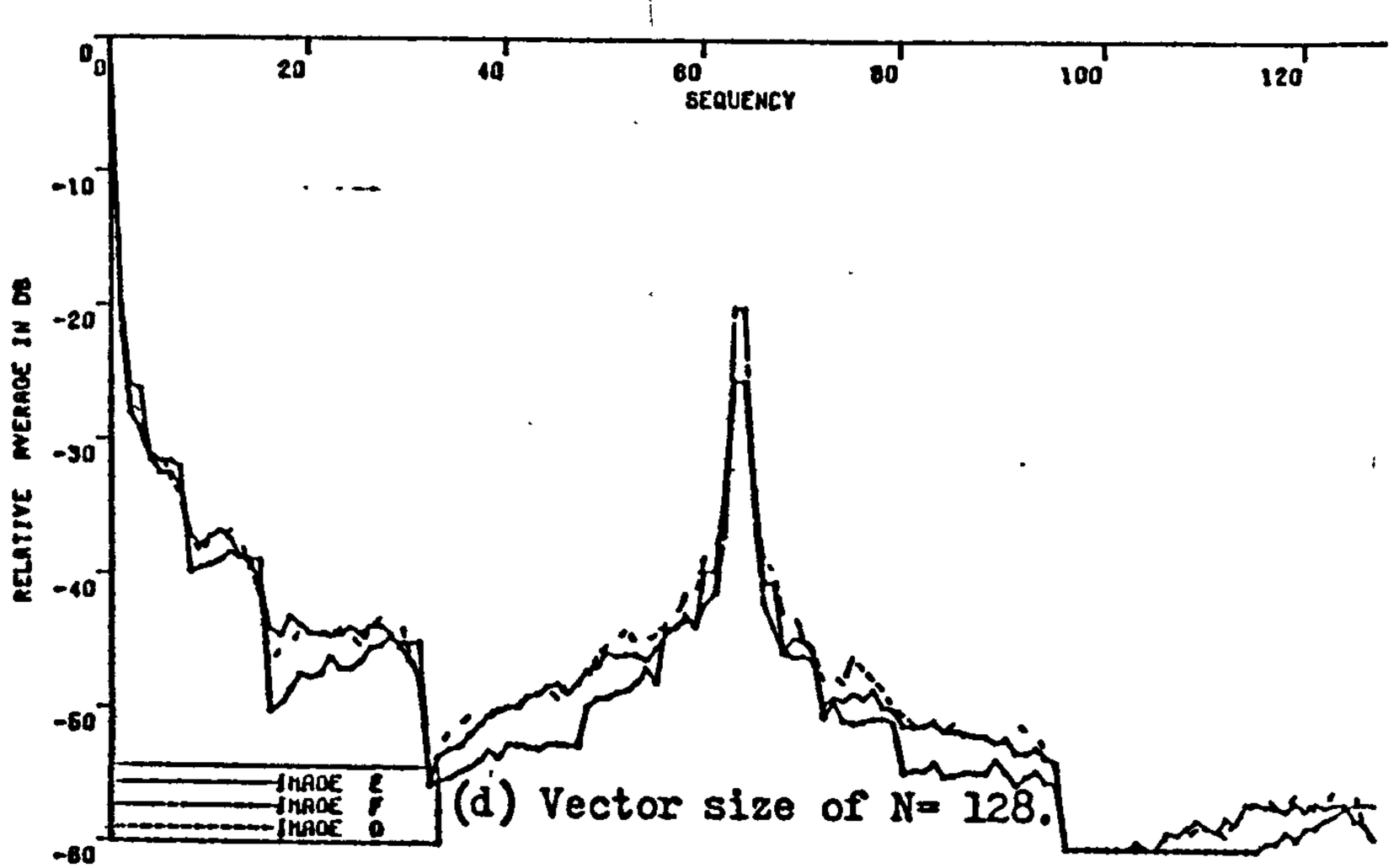
(a) Vector size of N= 16.



(b) Vector size of N= 32.



(c) Vector size of N= 64.



(d) Vector size of N= 128.

Figure (3.15). Relative averages of Direct transform coefficients at $4f_{sc}$ for different vector sizes.

Table (3.4). Levels of transform coefficient values. Direct transform, N=16.

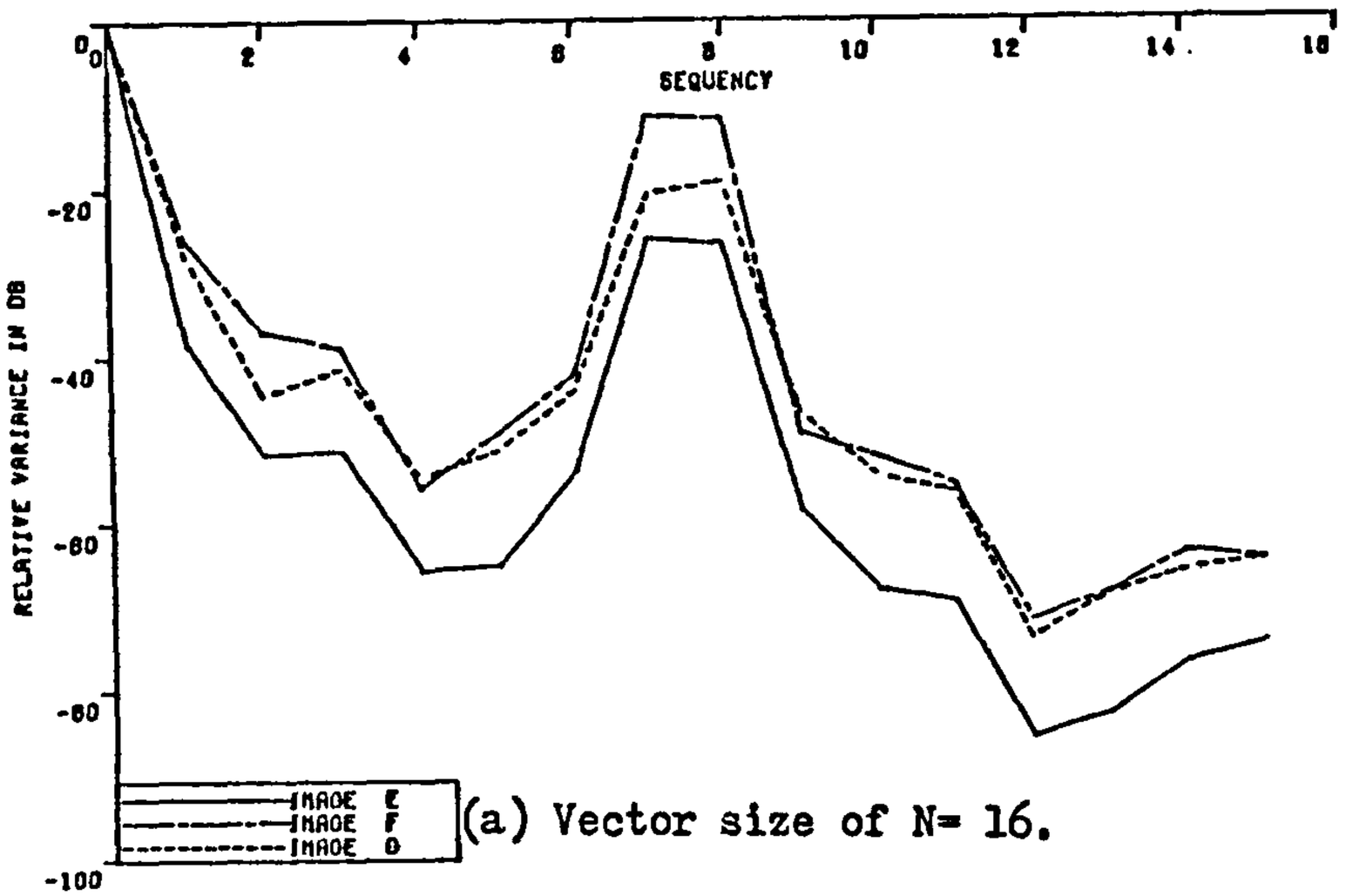
Levels of decrease than dc term.	Percent of coefficients with values within these levels.		
Image :	E	F	G
18 -- 24 dB.	-	12.5	-
24 -- 30 dB.	12.5	6.3	12.5
30 -- 36 dB.	6.3	6.3	6.3
36 -- 42 dB.	18.7	25	31.2
42 -- 48 dB.	31.2	18.7	18.7
48 -- 54 dB.	12.5	25	25
54 -- 60 dB.	12.5	-	-

Table (3.5). Levels of transform coefficient values. Direct transform, N=32.

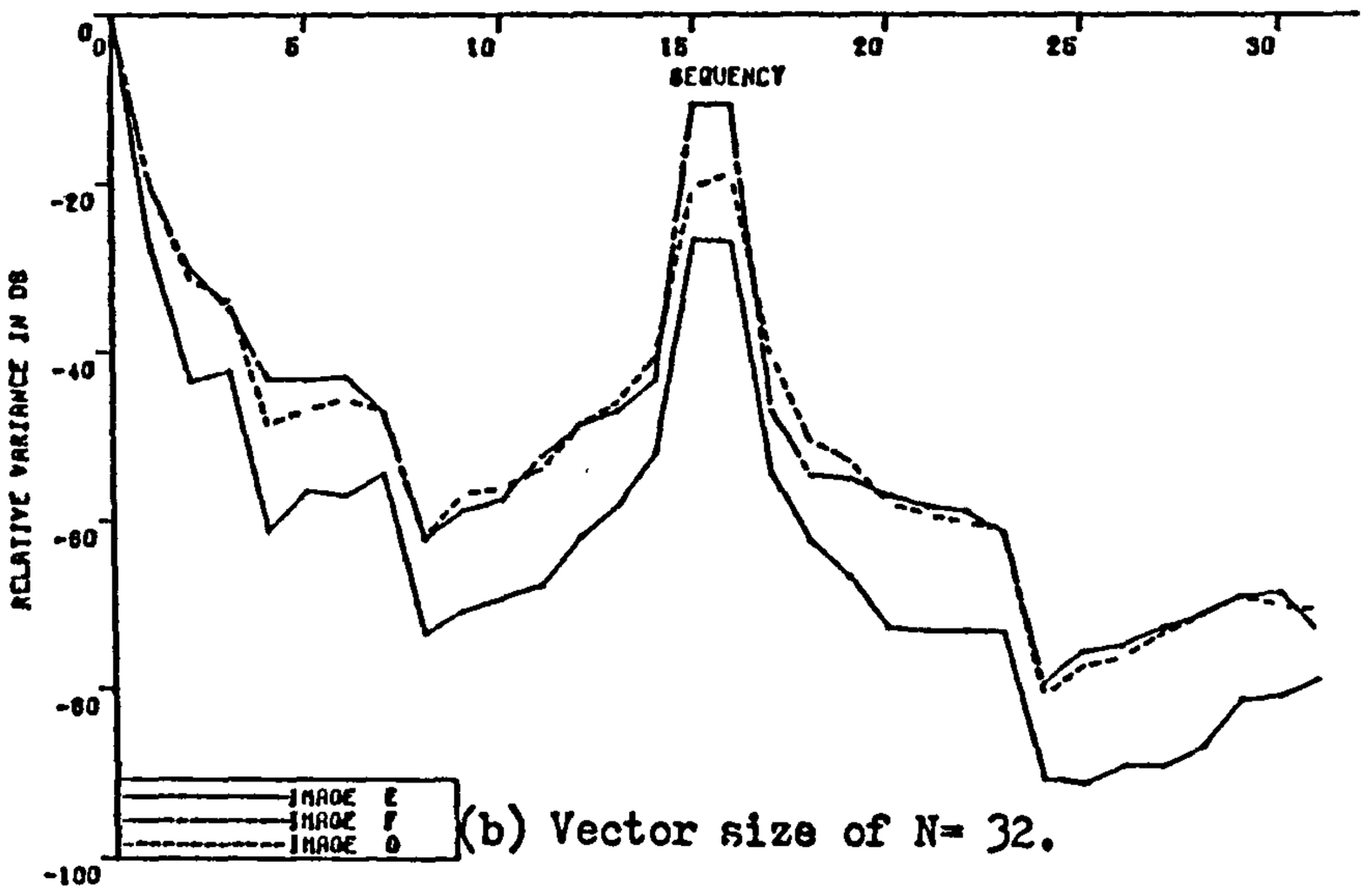
Levels of decrease than dc term.	Percent of coefficients with values within these levels.		
Image :	E	F	G
18 -- 24 dB.	-	6.3	-
24 -- 30 dB.	9.3	3.1	9.3
30 -- 36 dB.	6.3	6.3	6.3
36 -- 42 dB.	12.5	25	28.1
42 -- 48 dB.	25	31.3	28.1
48 -- 54 dB.	25	18.8	18.8
54 -- 60 dB.	18.8	6.3	6.3

Table (3.6). Levels of transform coefficient values. Direct transform, N=64.

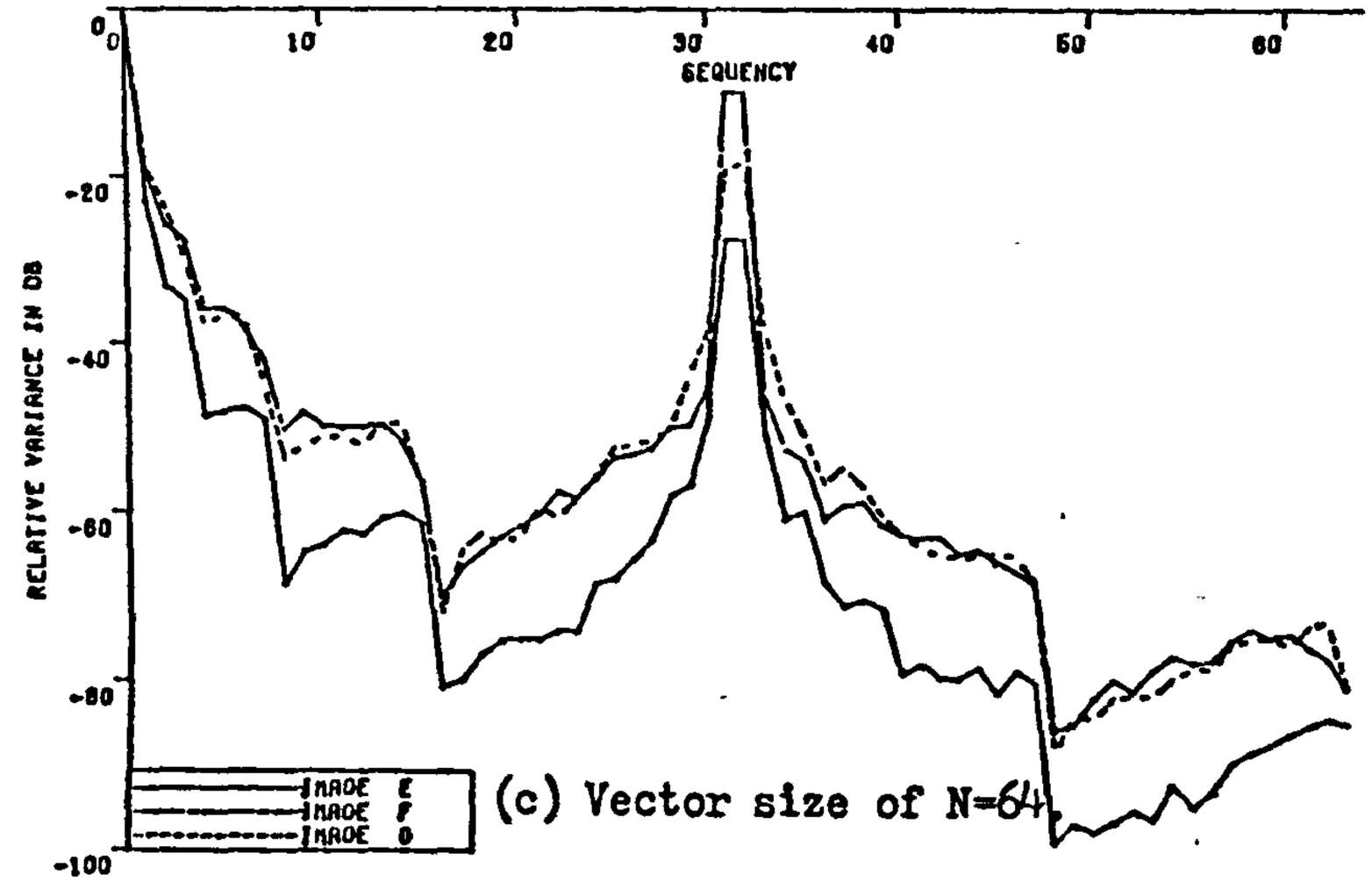
Levels of decrease than dc term.	Percent of coefficients with values within these levels.		
Image :	E	F	G
18 -- 24 dB.	1.6	3.1	-
24 -- 30 dB.	6.3	3.1	6.3
30 -- 36 dB.	-	6.3	7.8
36 -- 42 dB.	12.5	18.8	17.1
42 -- 48 dB.	25	25	28.1
48 -- 54 dB.	28.1	18.8	21.8
54 -- 60 dB.	15.6	23.4	17.1
60 -- 66 dB.	9.3	-	-



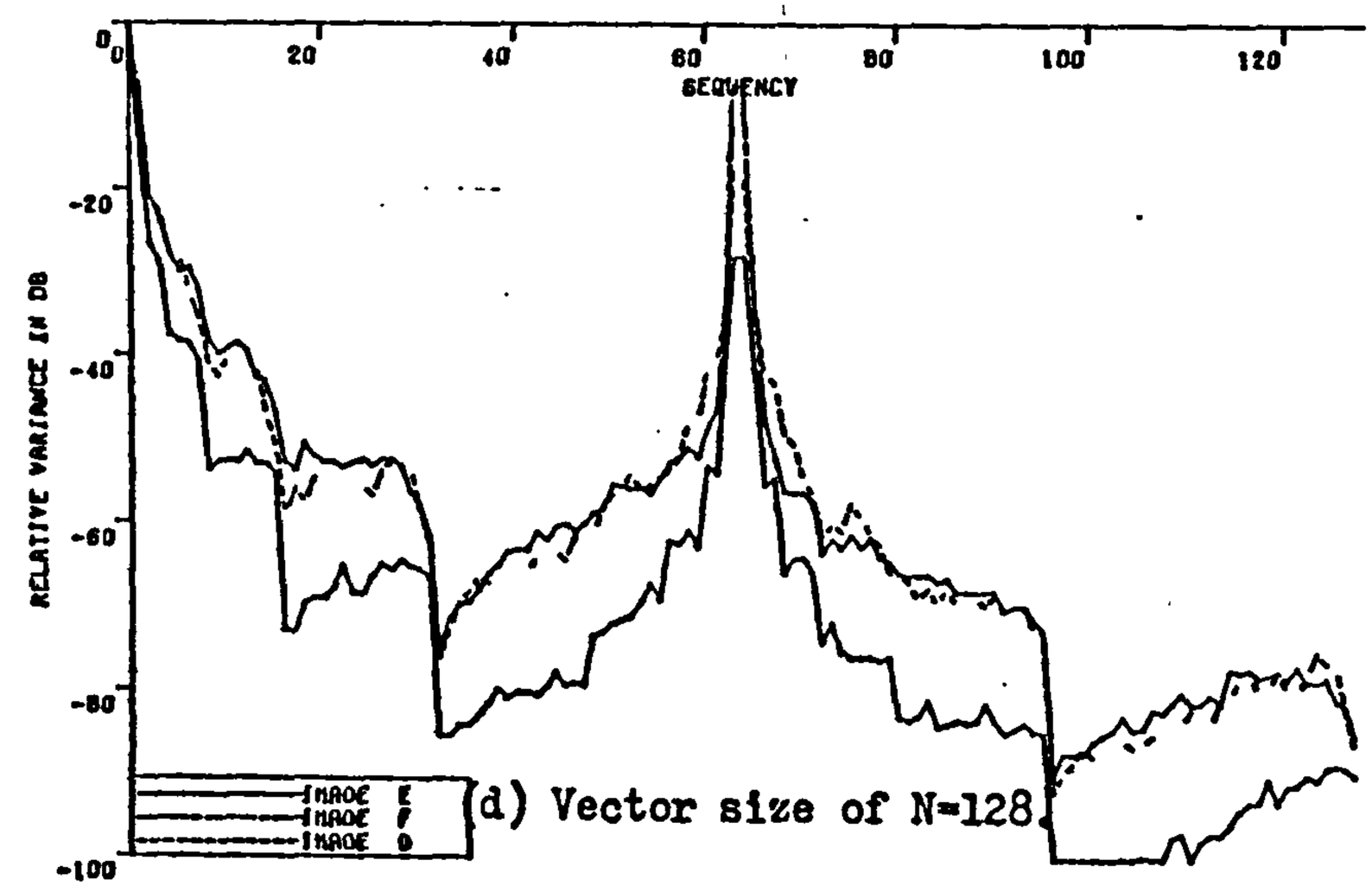
(a) Vector size of N= 16.



(b) Vector size of N= 32.



(c) Vector size of N=64.



(d) Vector size of N=128.

Figure (3.16). Relative variances of Direct transform coefficients at $4f_{sc}$ for different vector sizes.

3.2.1.2. Energy Packing Efficiency:

Table (3.7) shows the ac energy packing efficiency for a vector size of N= 16. In accordance with Section (3.2.1.1.), the table shows a rise in energy content of between (72 % - 87 %) due to the two mid-sequency terms alone. Figure (3.17) shows the energy packing efficiency for more vector sizes. From these graphs, the ac energy content of coefficients in (lower half of spectrum plus one),is nearly constant at about 98 % of the total ac energy, regardless of vector size or image category. For bigger transforms, as 128, the energy is much more spread among medium sequency coefficients close to the mid-sequency range. This suggests that a smaller vector size of N= 8, 16, or 32 will be more efficient for the $4f_{sc}$ direct transform. In such a case, only two or four mid-sequency coefficients may preserve most of the ac energy, while the dc term will preserve all the dc energy.

For a quick comparison among different vector sizes, Table (3.8) shows the ac energy content in the two mid-sequency terms for different sizes and different image categories. From the table, the decrease in ac energy concentration as the vector sizes increase is apparent. It is also clear from the table and Figure (3.17) that this decrease of concentration is also strongly dependent on image category, especially at bigger vector sizes.

Table (3.7). Energy Packing Efficiency for $4f_{sc}$ Direct tarnsform. Vector size N= 16.

Image categ.	E	F	G
Up to coeff. (included).	ac energy content %	ac energy content %	ac energy content %
1	0	0	0
2	9.2	5.9	10.2
3	11.2	7.6	13.5
4	13.6	9.0	16.7
5	14.2	9.3	17.9
6	15.0	10.0	19.7
7	16.9	11.3	22.8
8	57.1	55.0	58.6
9	97.0	98.3	94.2
10	98.3	99.0	96.2
11	98.9	99.4	97.6
12	99.4	99.7	98.7
13	99.5	99.7	99.0
14	99.6	99.8	99.3
15	99.8	99.9	99.7
16	100	100	100

Table (3.8). Energy content in the two mid-sequency coefficients in Direct transform.

Vector size.	Mid-seq. orders. (seq. + 1).	ac energy content % in image E	ac energy content % in image F	ac energy content % in image G
16	8,9	80.1	87.0	71.4
32	16,17	67.0	78.3	58.2
64	32,33	49.0	69.4	45.2
128	64,65	26.3	56.6	28.8

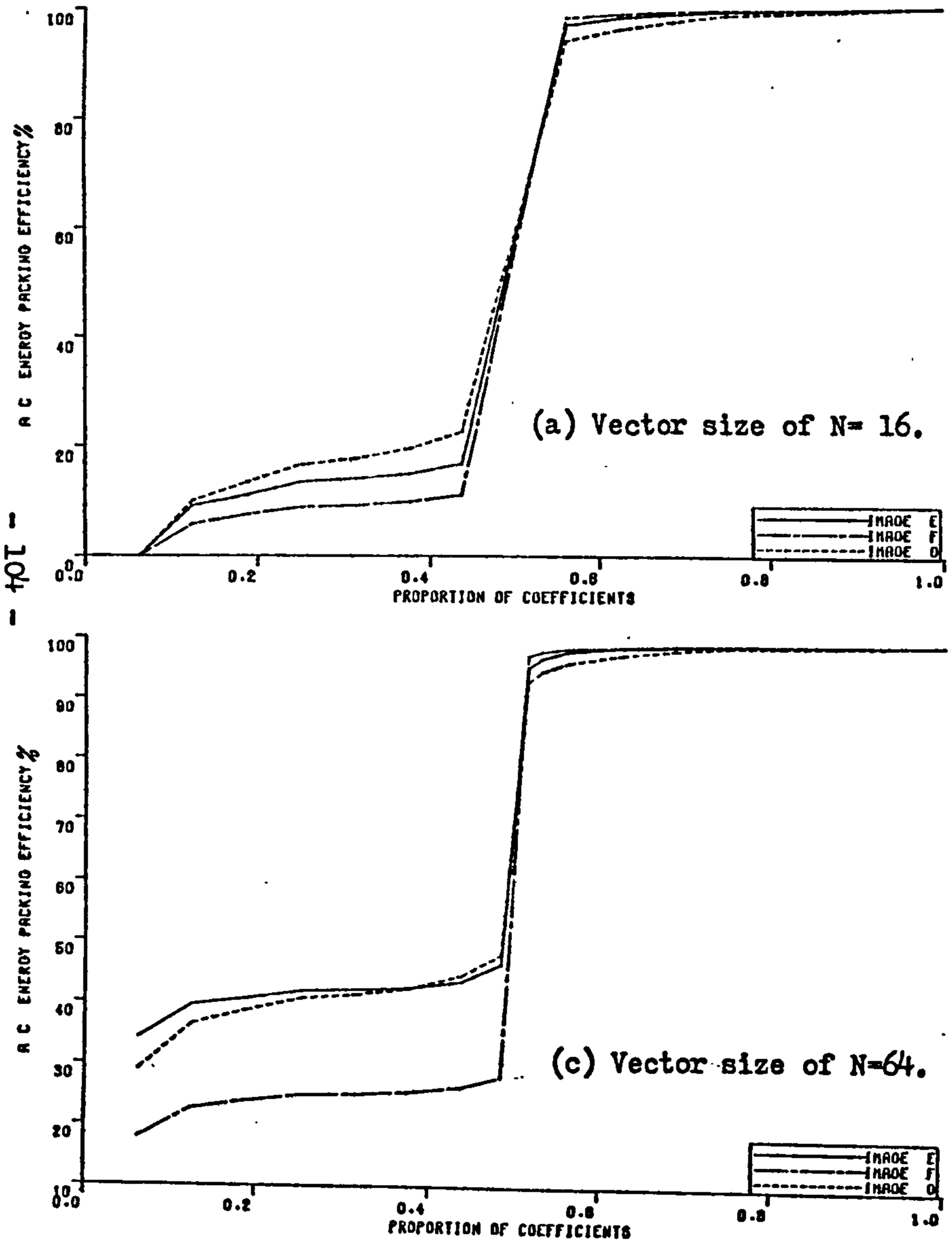


Figure (3.17). AC energy packing efficiency of Direct transform at $4f_{sc}$ for different vector sizes.

3.2.2. Laced Transform at $4f_{sc}$:

This transform is applied in a similar way to the Laced transform in case of $3f_{sc}$. As with sampling in Direct transform for $4f_{sc}$, the phases are chosen at $\pi/2$, π , $3\pi/2$, and 2π (although other phases could be chosen). Figure (3.18) shows the composite signal for several subcarrier intervals. Samples are indicated on the wave shape as A's, B's, C's, and D's.

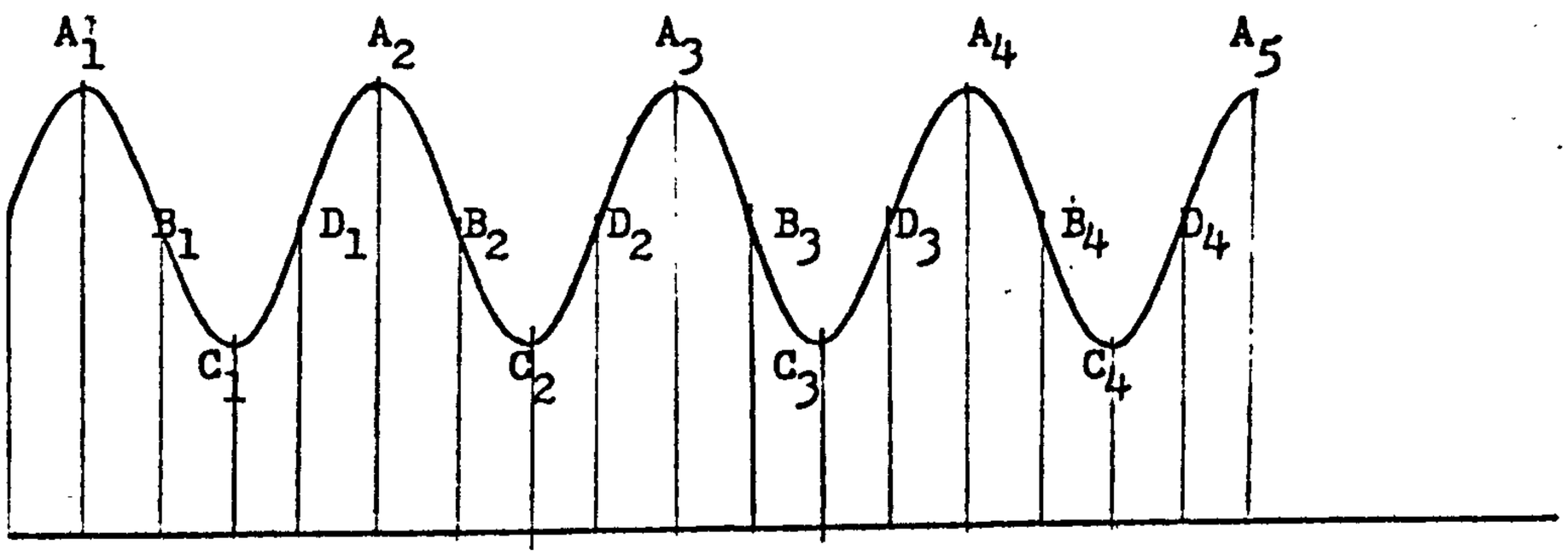
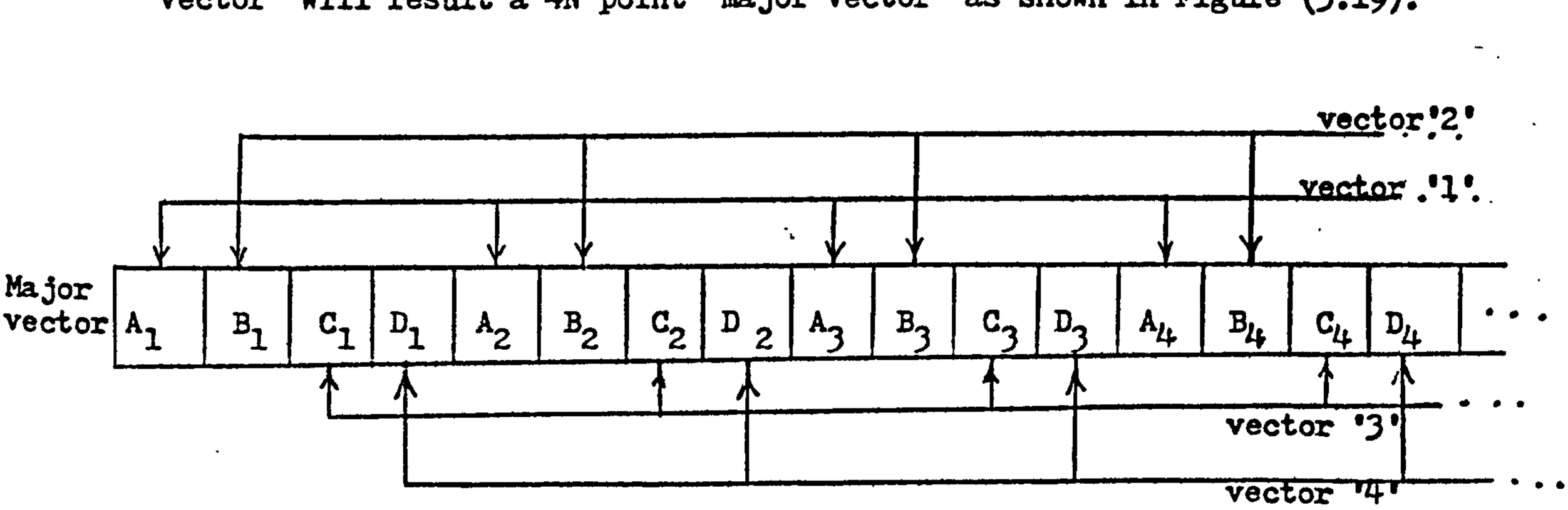


Figure (3.18). Composite colour signal sampled at $4f_{sc}$.

Taking N subcarrier cycles and grouping all the samples in one 'vector' will result a $4N$ point 'major vector' as shown in Figure (3.19).



Vectors '1', '2', '3', and '4' are the 'minor' vectors.

Figure (3.19). Major vector and its four 'minor' vectors of a $4f_{sc}$ laced transform.

Starting with sample A_1, and taking each fourth sample after that, the first 'minor' vector will then be extracted as:

$A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \ldots\ldots\ldots\ldots\ldots A_N$

Similarly, for B, C, and D samples, the corresponding vectors are:

$B_1 \quad B_2 \quad B_3 \quad B_4 \quad B_5 \ldots\ldots\ldots\ldots\ldots B_N$

$C_1 \quad C_2 \quad C_3 \quad C_4 \quad C_5 \ldots\ldots\ldots\ldots\ldots C_N$

$D_1 \quad D_2 \quad D_3 \quad D_4 \quad D_5 \ldots\ldots\ldots\ldots\ldots D_N$

Like in the case in $3f_{sc}$, each vector consists of N samples exactly one subcarrier interval apart from each other. This will make the samples in a minor vector highly correlated, resulting a high degree of decorrelation in the transform domain. Transforming is the same as in monochrome and $3f_{sc}$ sampled colour signals.

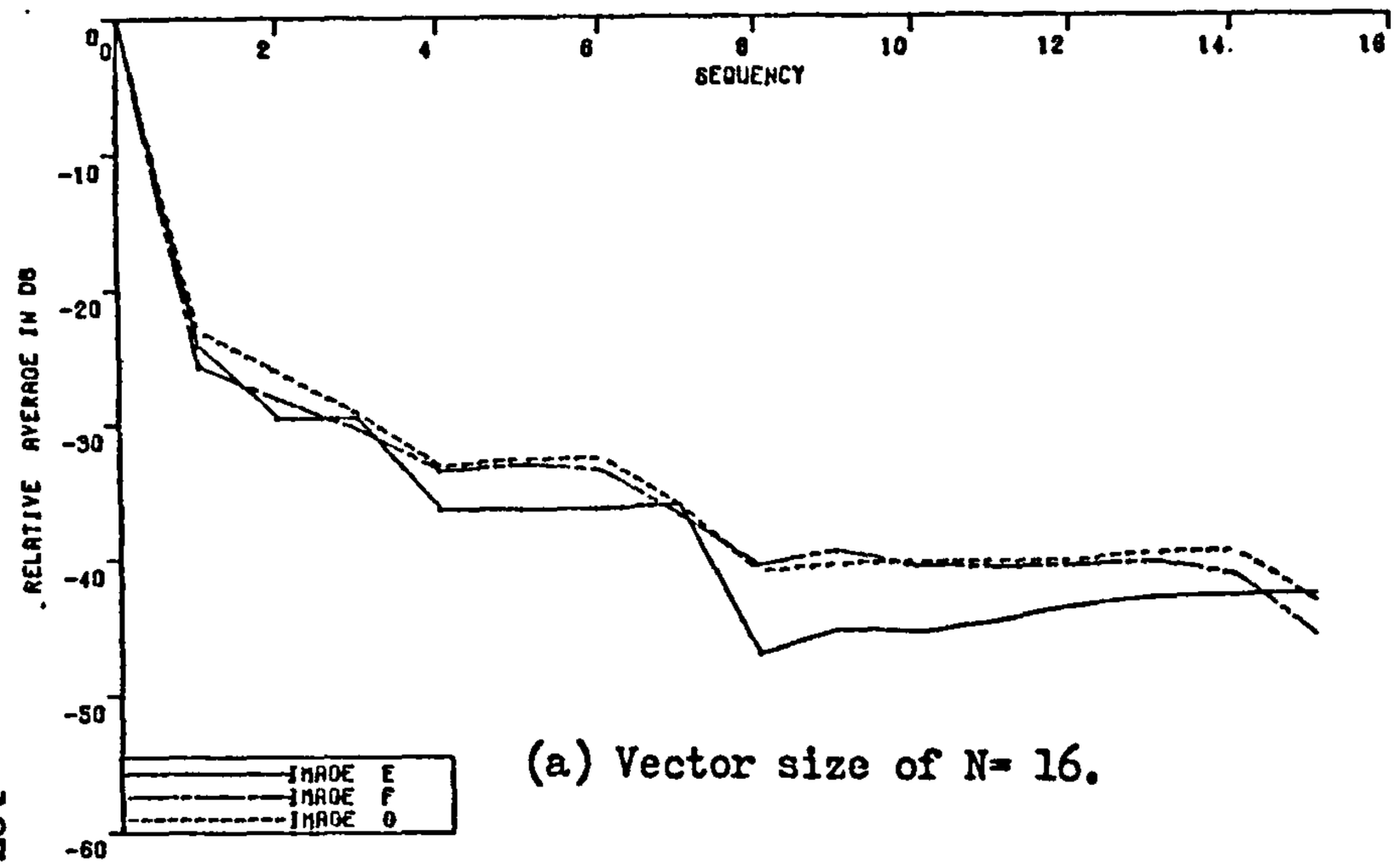
Actually, no noticeable differences have been noticed in transform domain characteristics as compared with $3f_{sc}$ Laced transform. This is expected since the samples in each vector in both cases represent the same data at different sampling points. Therefore, only some of the graphs will be shown here without comments or explanations. All the graphs have same definitions and variables as the corresponding ones in the case of $3f_{sc}$ laced transform.

3.2.2.1. Averages and Variances of Coefficients:

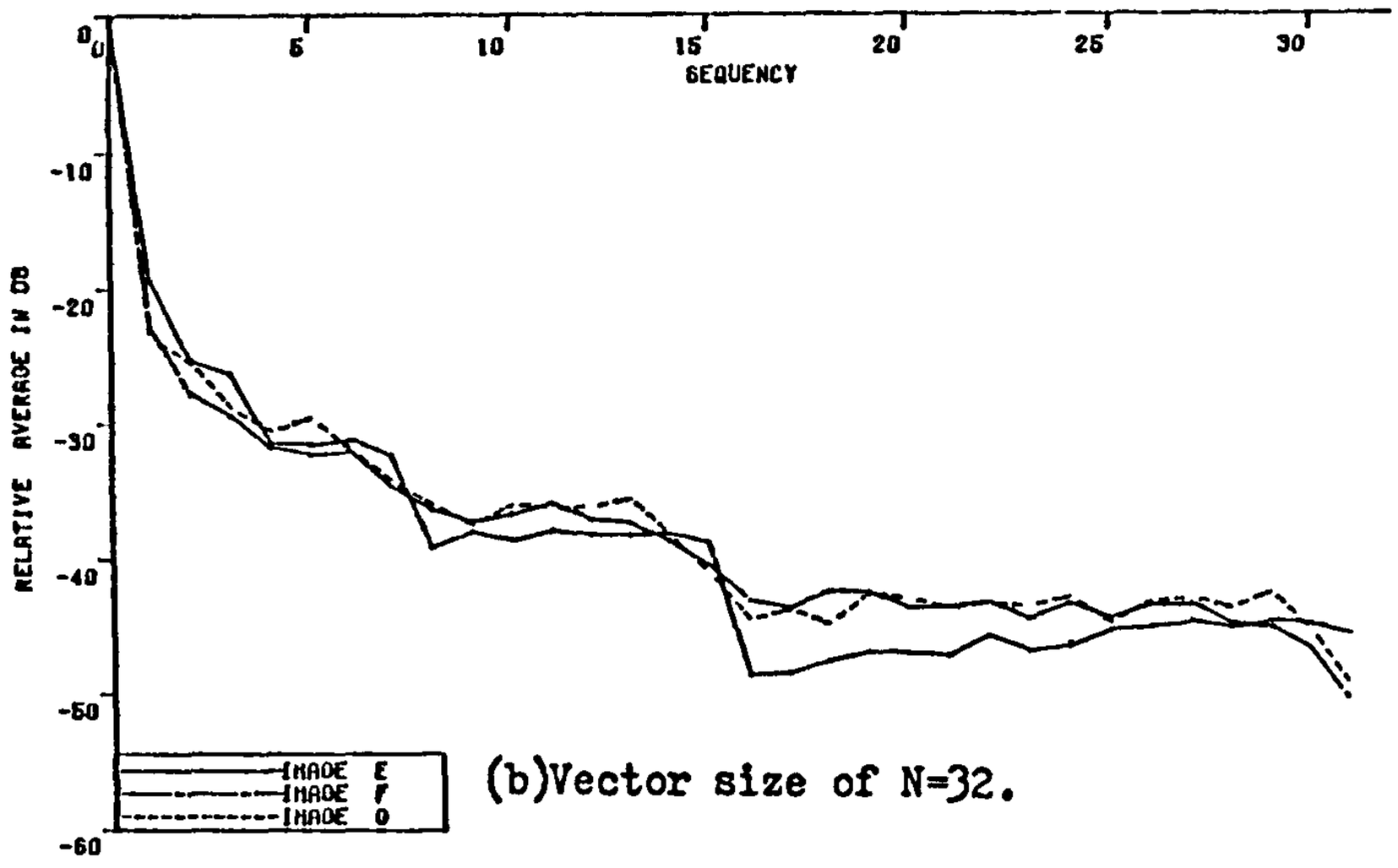
Figure (3.20) shows some relative values of coefficients for different cases. Figure (3.21) shows relative variances for some cases.

3.2.2.2. Energy Packing Efficiency:

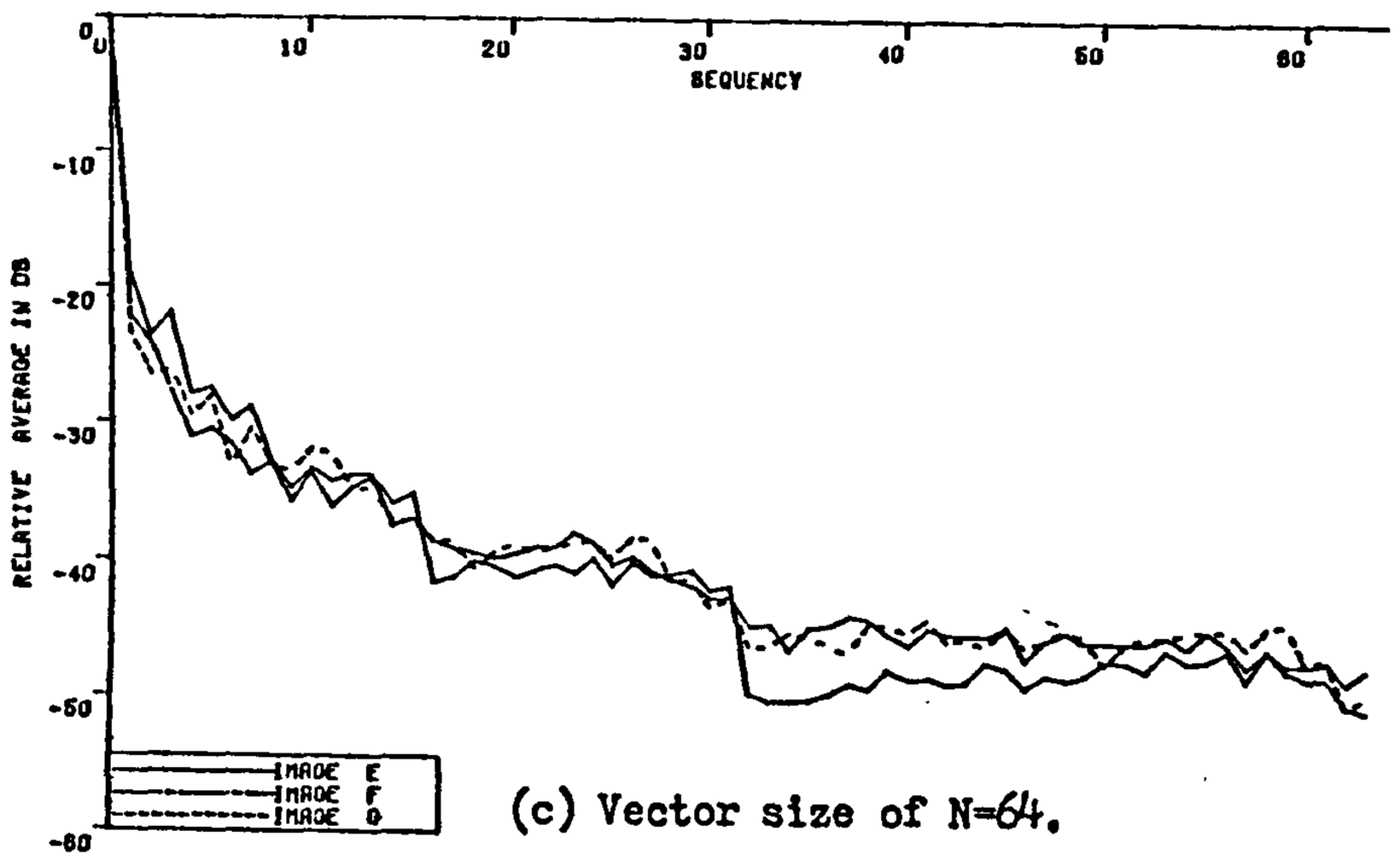
Figure (3.22) shows energy packing efficiency for some vector sizes and image categories. General shapes are nearly the same as in the case of $3f_{sc}$ laced transform.



(a) Vector size of N= 16.

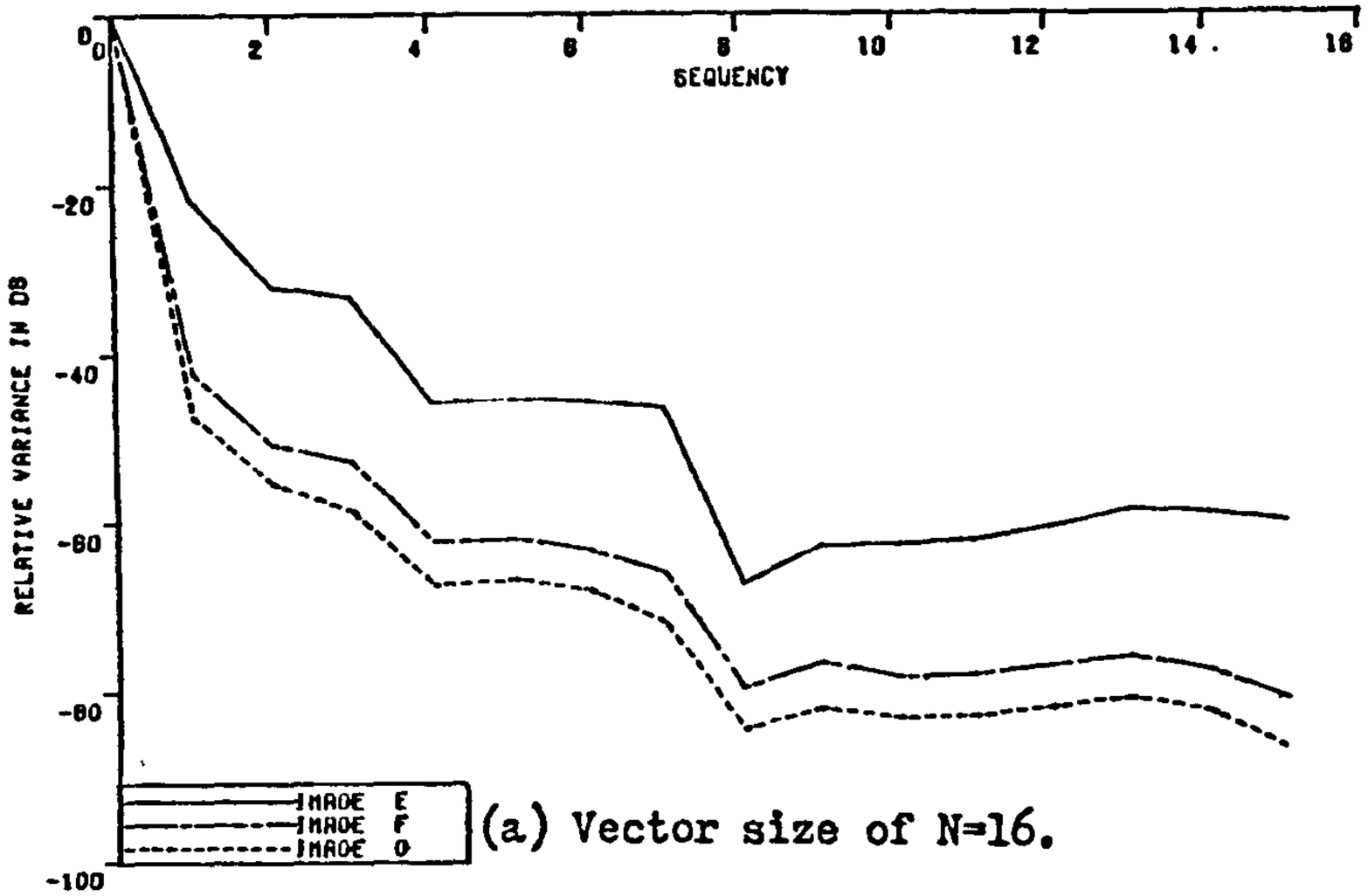


(b)Vector size of N=32.

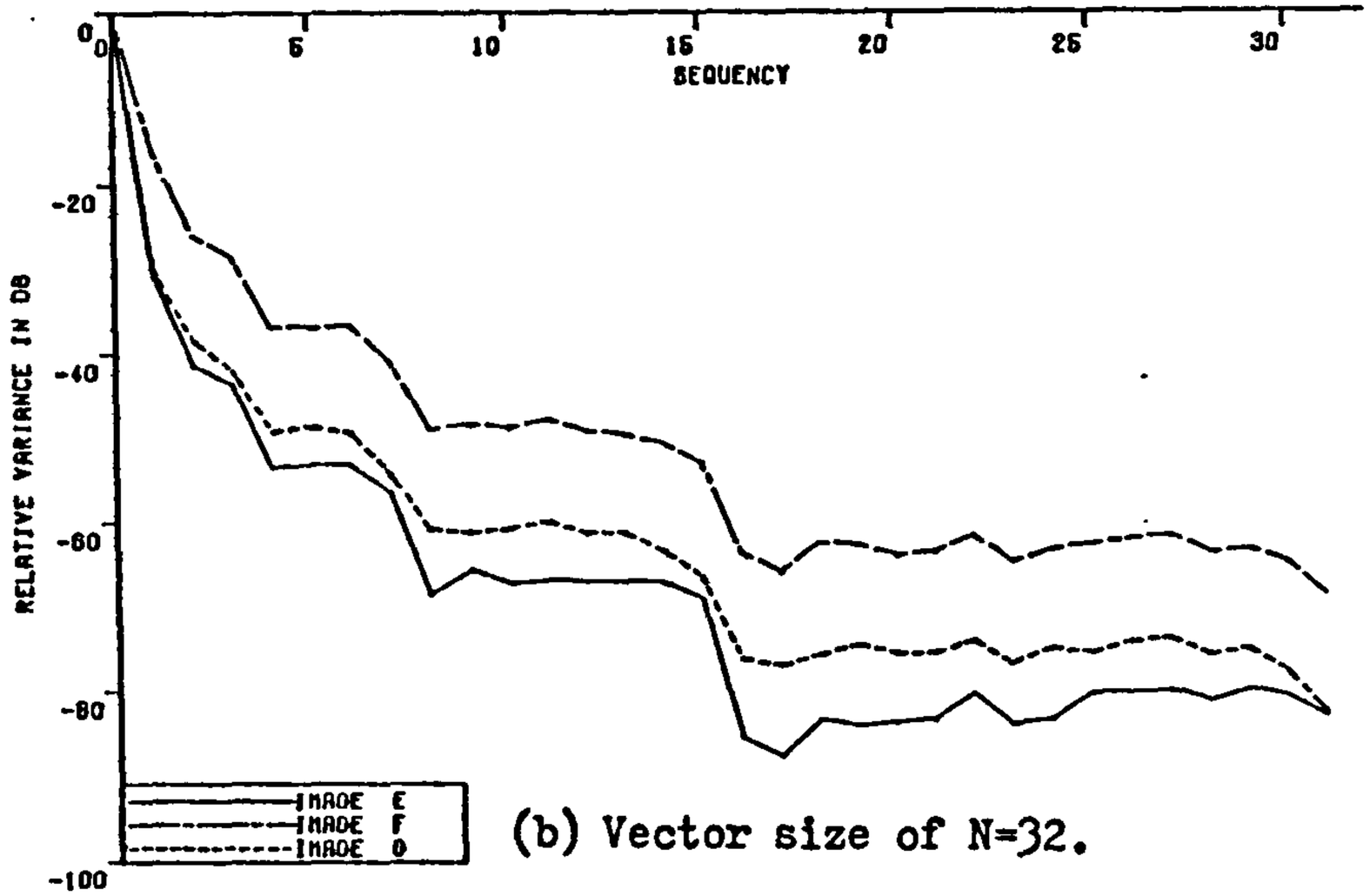


(c) Vector size of N=64.

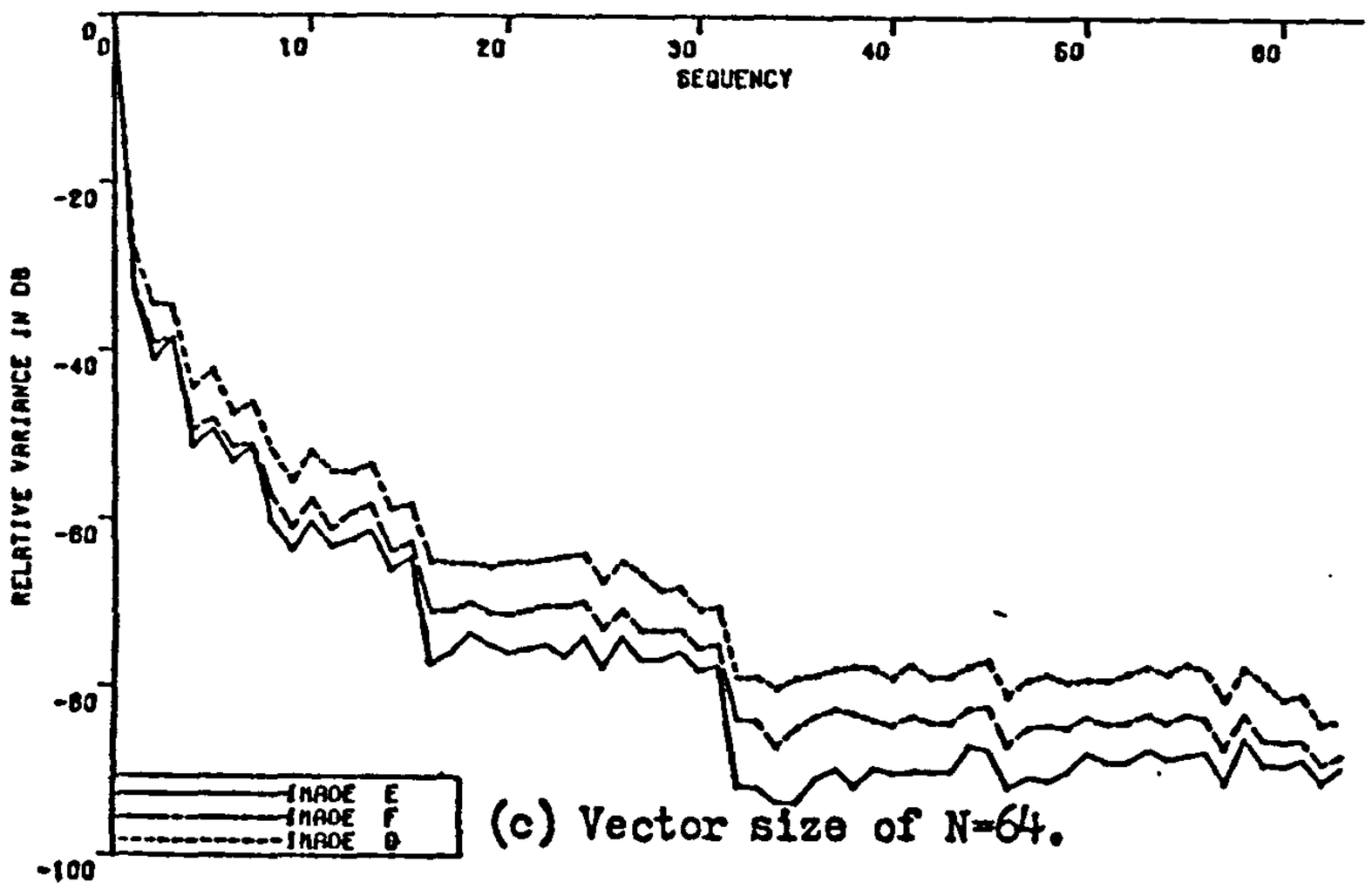
Figure (3.20). Relative averages of Laced transform coefficients at $4f_{sc}$ for different vector sizes.



(a) Vector size of N=16.

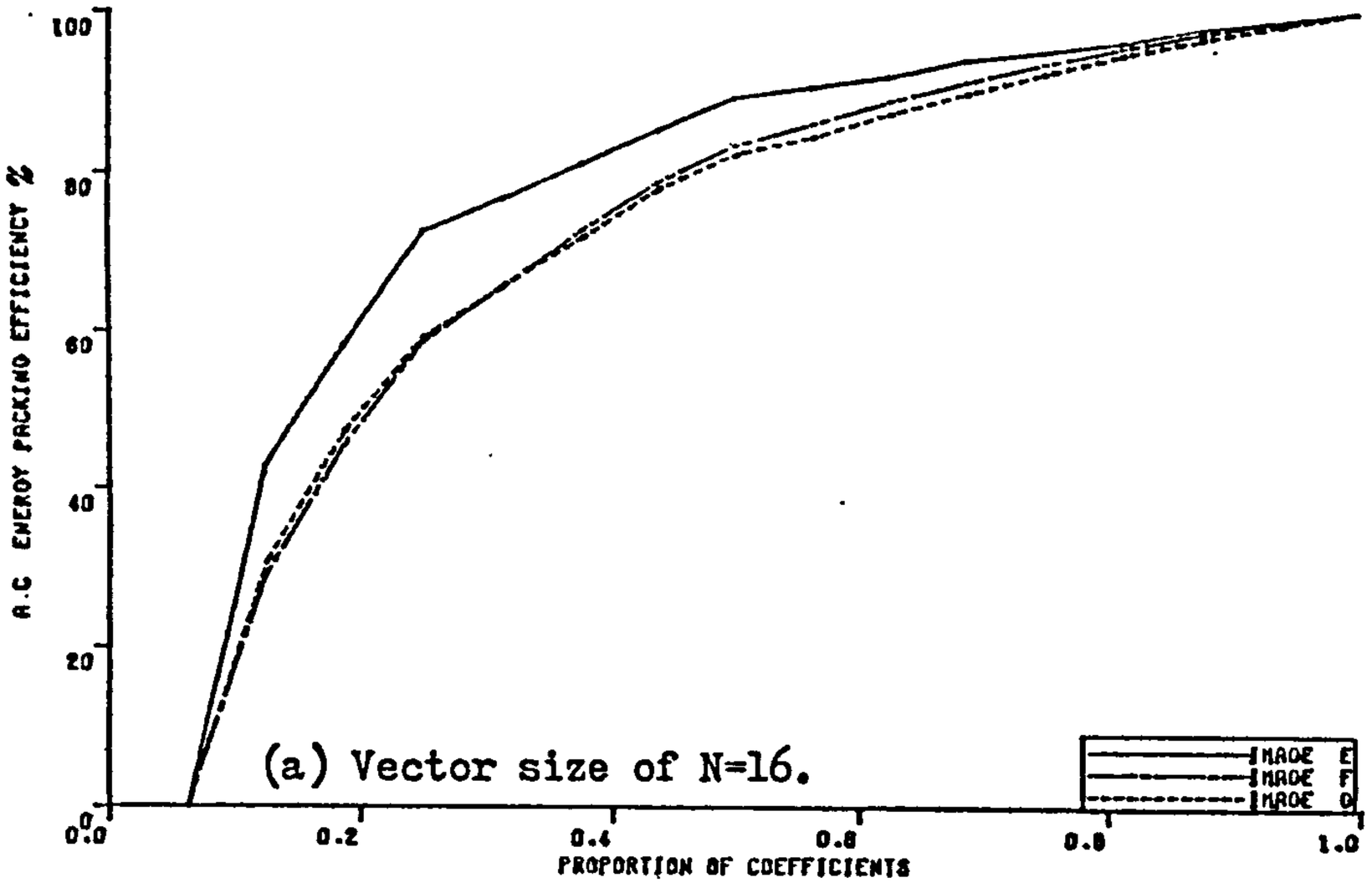


(b) Vector size of N=32.

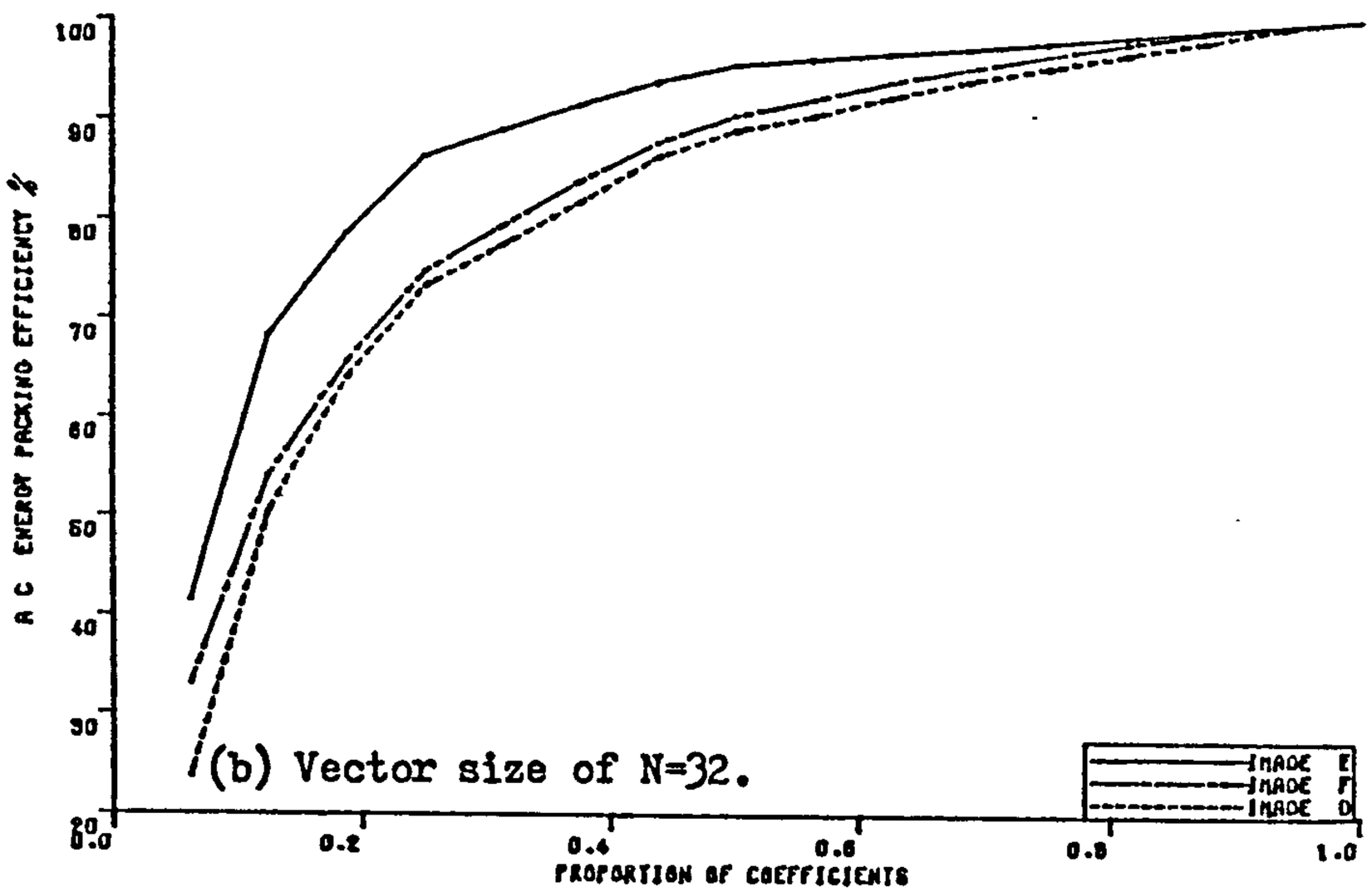


(c) Vector size of N=64.

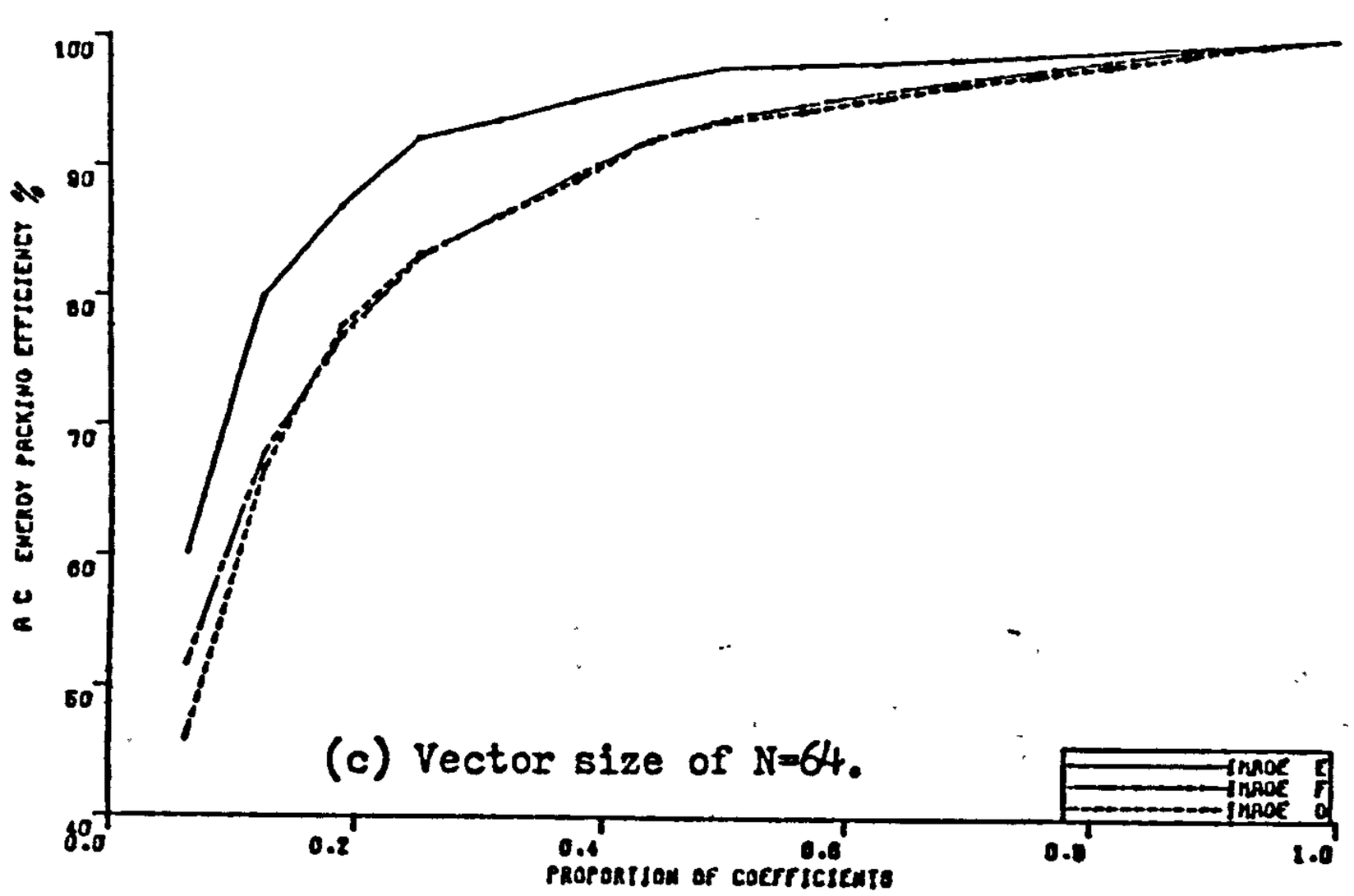
Figure (3.21). Relative variances of Laced transform coefficients at $4f_{sc}$ for different vector sizes.



(a) Vector size of N=16.



(b) Vector size of N=32.



(c) Vector size of N=64.

Figure (3.22). AC energy packing efficiency of Laced transform at $4f_{sc}$ for different vector sizes.

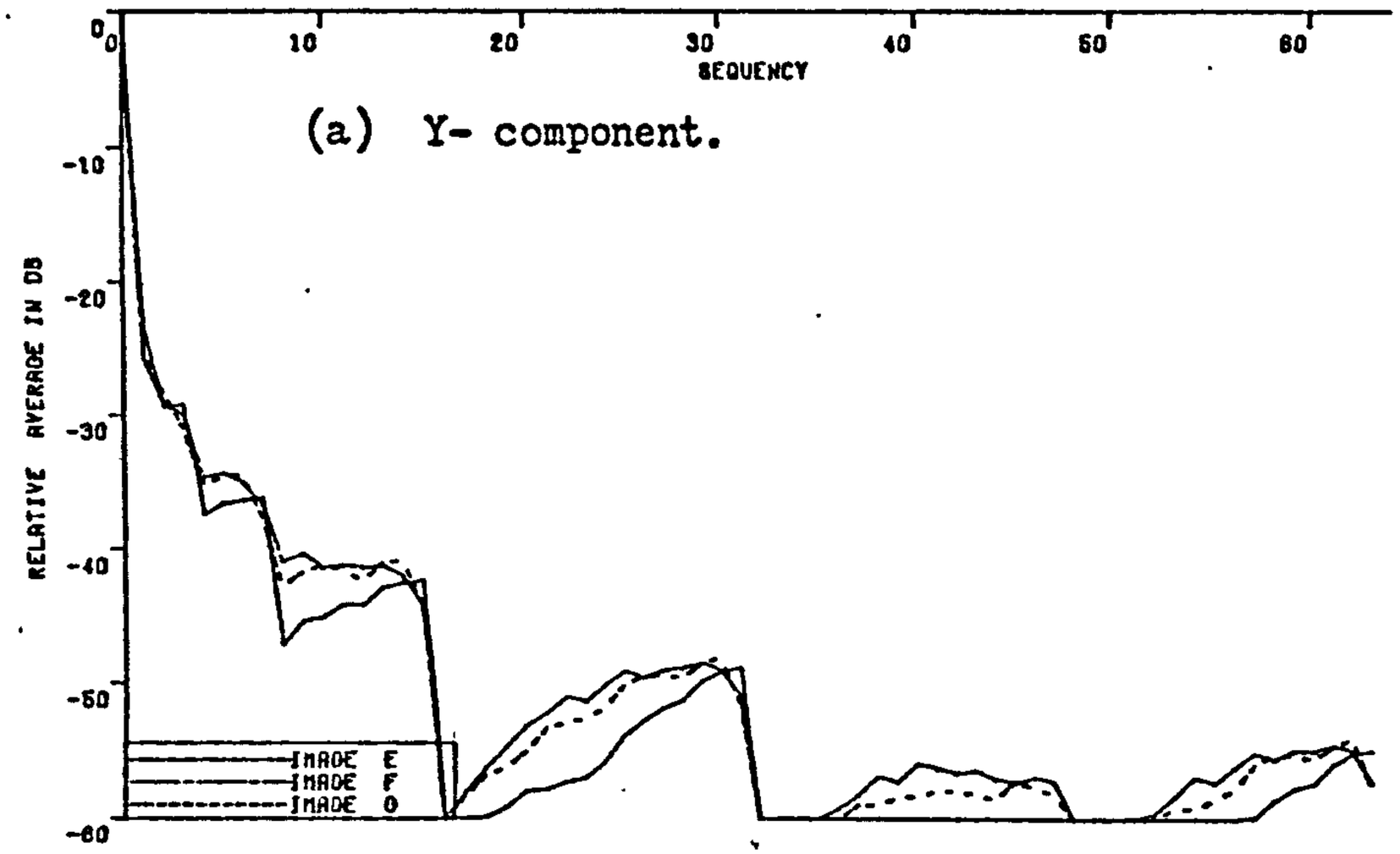
3.2.3. Components Transform at $4f_{sc}$:

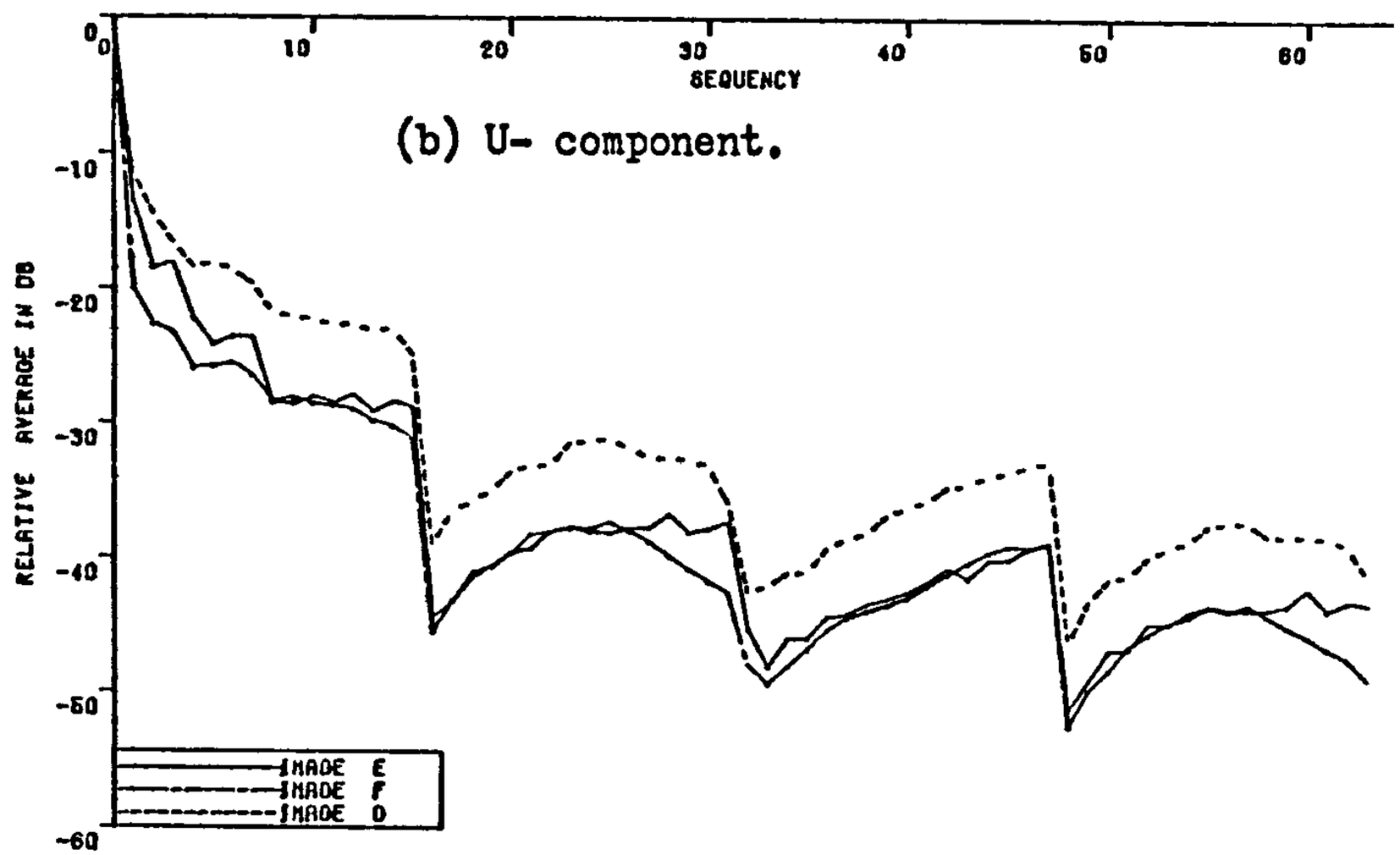
Each of the three components in the spatial domain is transformed separately. After coding, transmission, and decoding, inverse transformation will result in the original vectors. These vectors are then coded in the usual way to get the composite colour signal. It has been mentioned in Section (3.1.3.) that the total number of coefficients will be three times the vector size in case of $3f_{sc}$ sampling components transform. Similarly, of course, in the case of $4f_{sc}$ sampling, this number will be four times the vector size. However, due to the already large number of samples in a $4f_{sc}$ signal, the increase in number of coefficients in this components transform will be substantial.

3.2.3.1. Values and Variances of Coefficients:

Figure (3.23) shows the relative average values for the three components Y, u, and v for the three image categories in example vector size of N= 64. From these graphs, Y component is nearly similar to the corresponding $3f_{sc}$ transform. Chrominance components u, and v, are different in their characteristics from the $3f_{sc}$ transform. Generally, values of many coefficients are much less than those of the $3f_{sc}$ case. A major difference also is the existence of some sort of 'bands', in both u,and v, components. In addition to the abruptancy for the lowest half sequency, the upper half (subvector) has two distinctive 'bands' of values. These bands are at orders (33- 48) and (49- 64). The same applies equally well to all image categories, with very slight, or no differences. One potential advantage of this characteristic is the possibility that a multizonal coding scheme can be used. In such a scheme, the few coefficients which have the larger values may be taken to represent the chrominance energy. For the present example, coefficients groups like (24- 32), (40- 48), and (57- 64) may be selected to represent the medium- and high sequency subvectors.

Figure (3.24) shows relative variances in the example of Figure (3.23). Variances have nearly same characteristics as the average values.





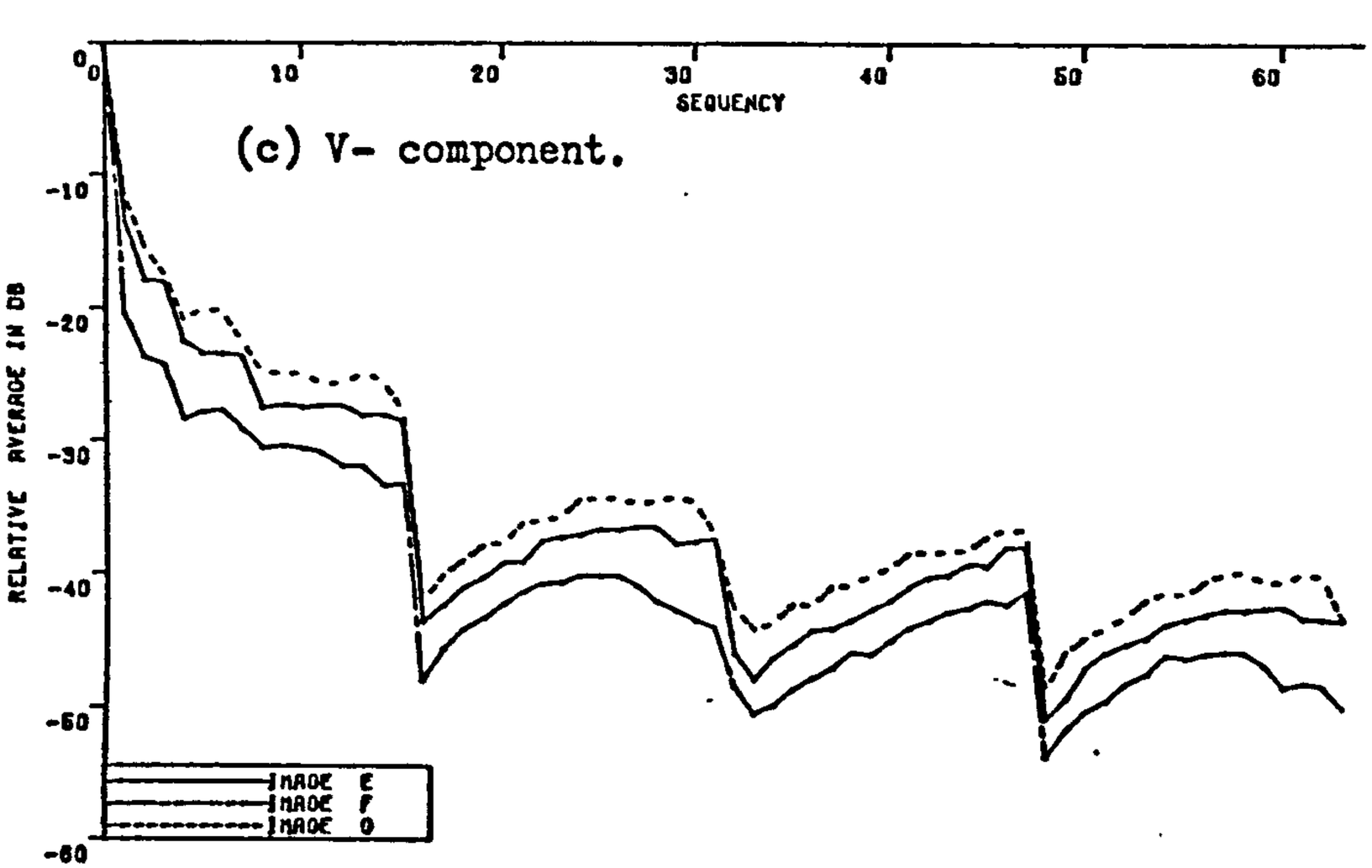
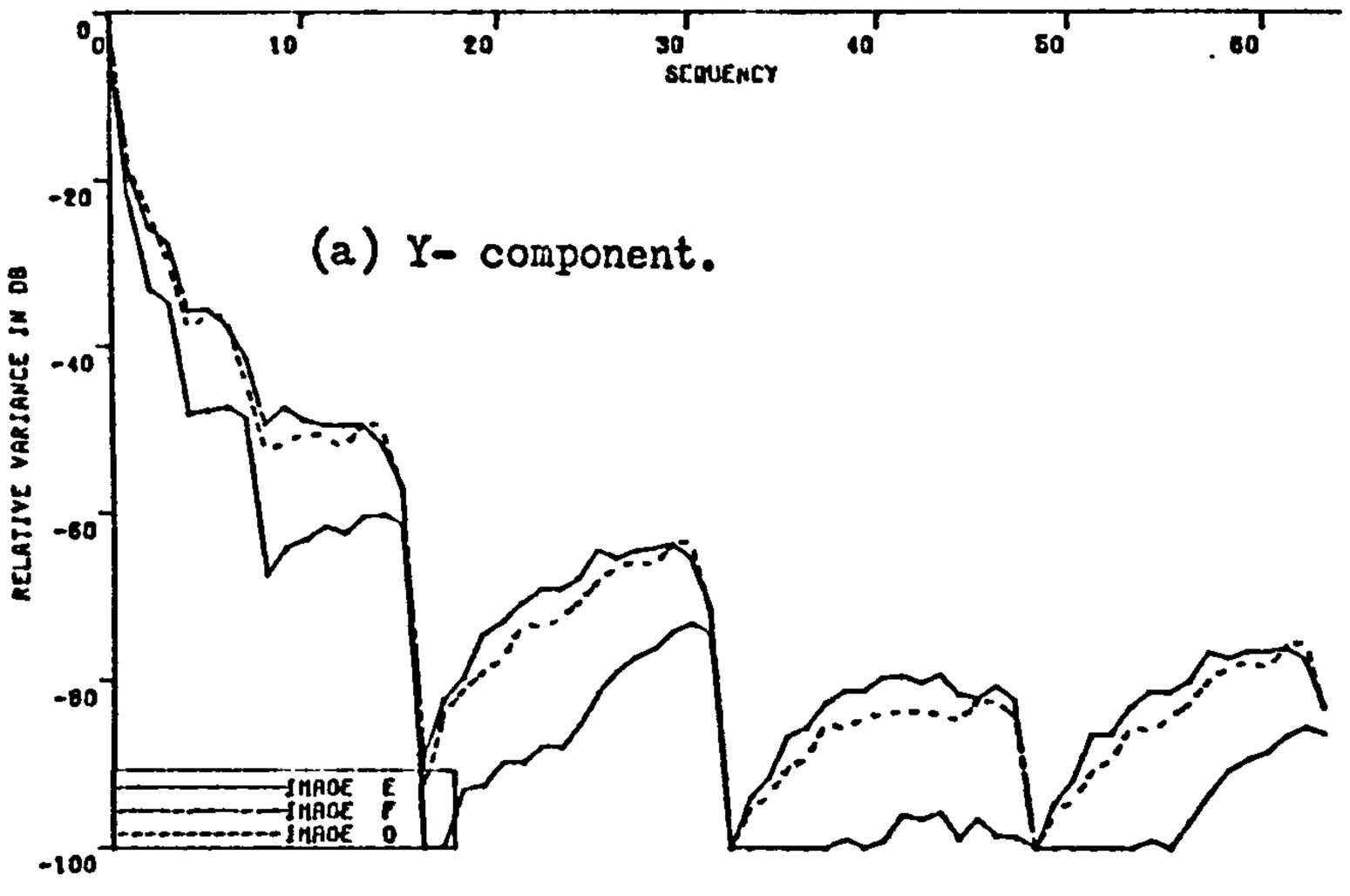
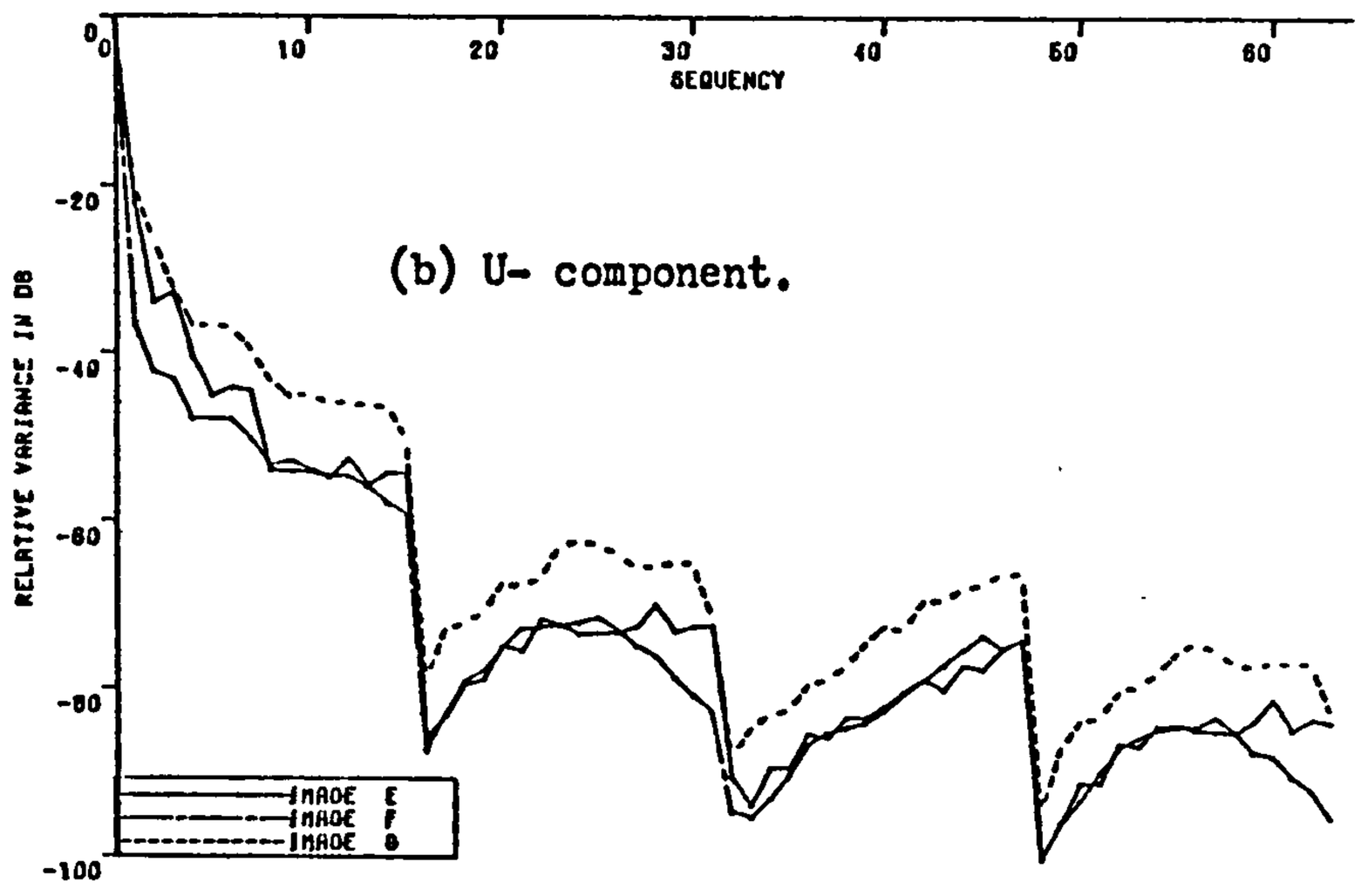


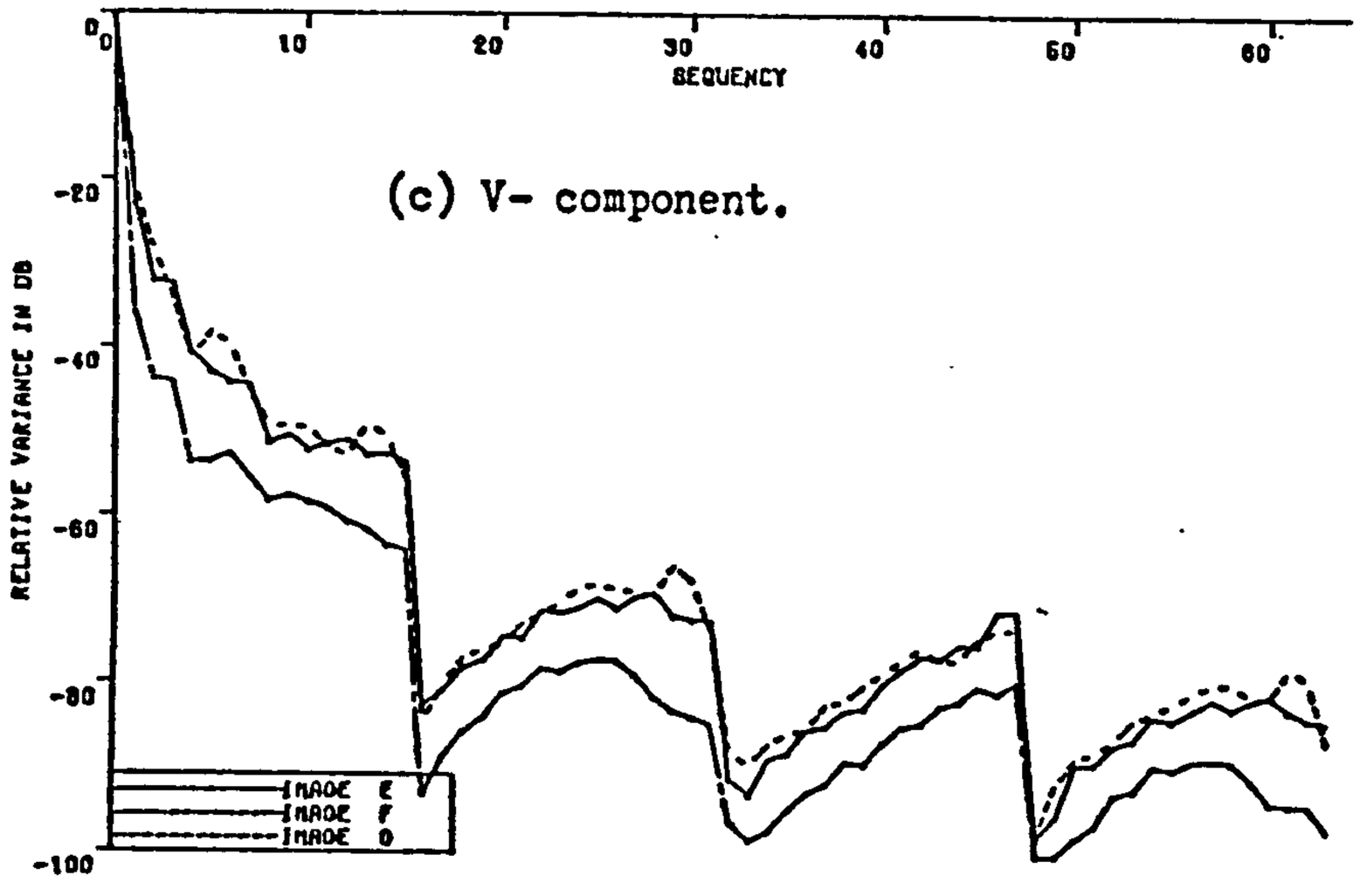
Figure (3.23). Relative averages of Components transform coefficients at $4f_{sc}$ for a vector size of N=64.



(a) Y- component.



(b) U- component.



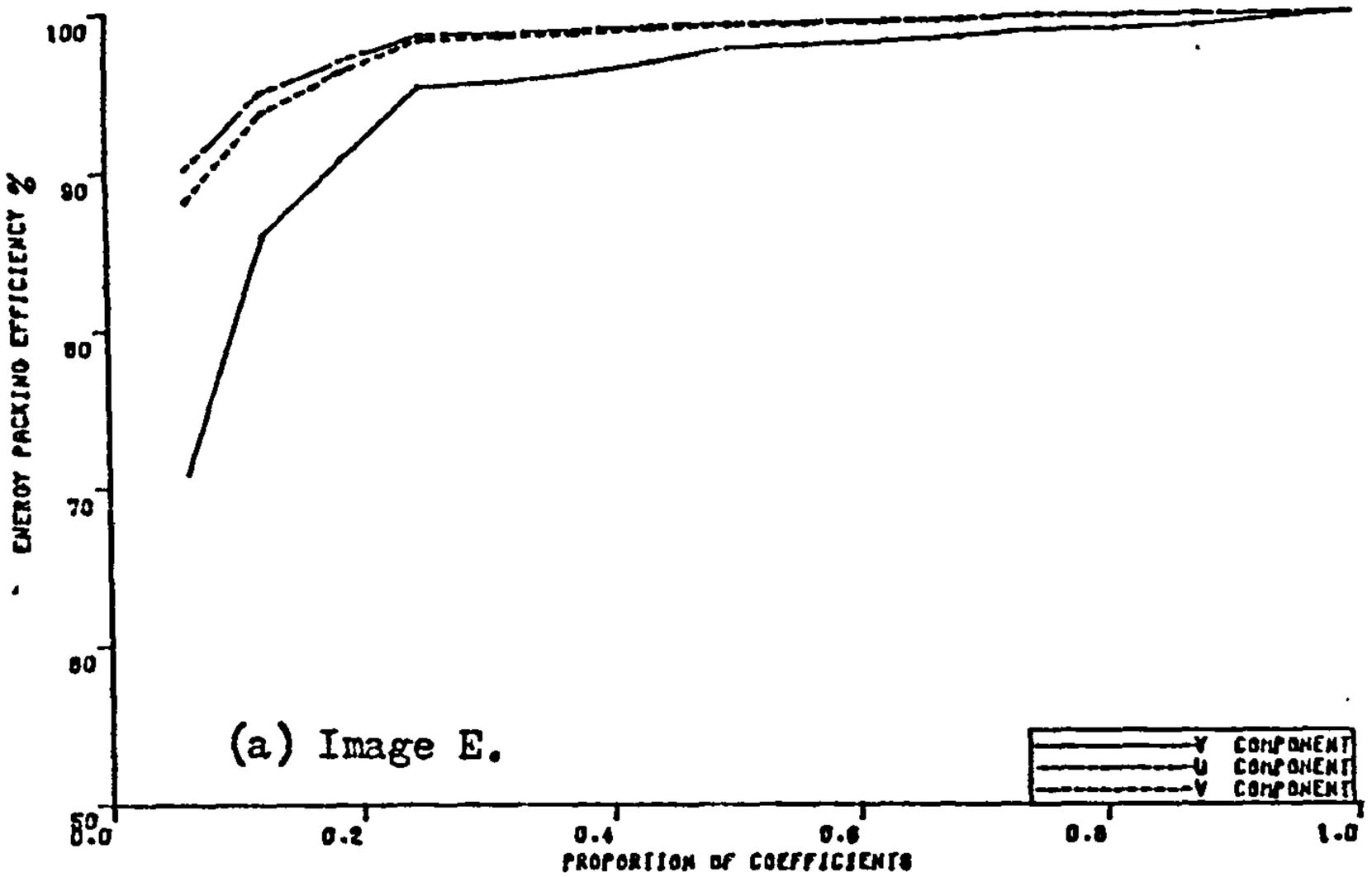
(c) V- component.

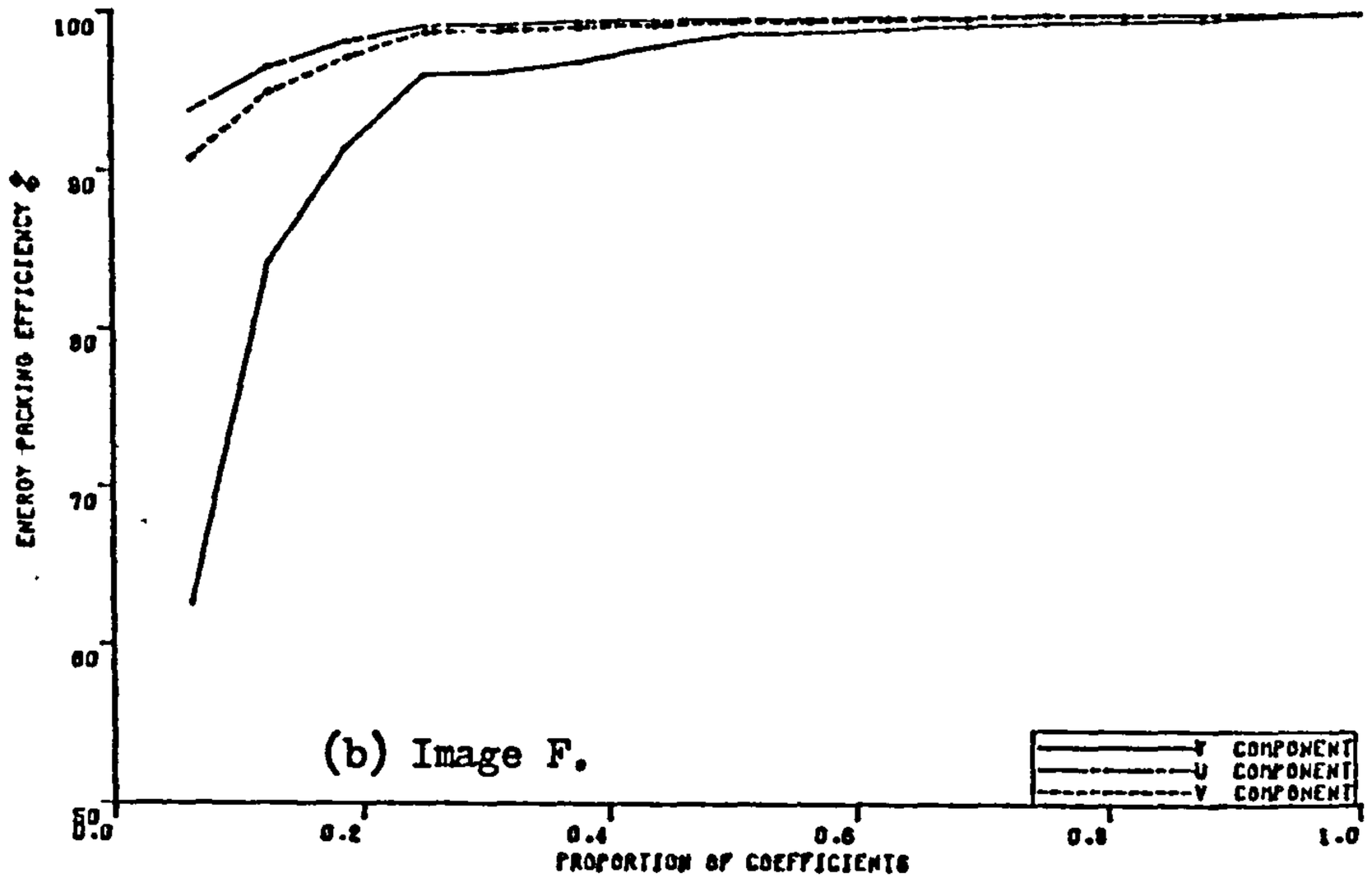
Figure (3.24). Relative variances of Components transform coefficients at $4f_{sc}$ for a vector size of N=64.

3.2.3.2.Energy Packing Efficiency:

Figure (3.25) shows the energy packing efficiency for different components transforms and different image categories. As in the case of $3f_{sc}$, for the Y component, only the ac energy is calculated, while for u,and v, components all energy including that of first coefficient in each case is considered, (Section 3.1.3.2.).

From all these graphs, it is clear that ac energy packing efficiency for Y component transform is always the poorest. For u, and v, efficiency is higher and they are very slightly different from each other.





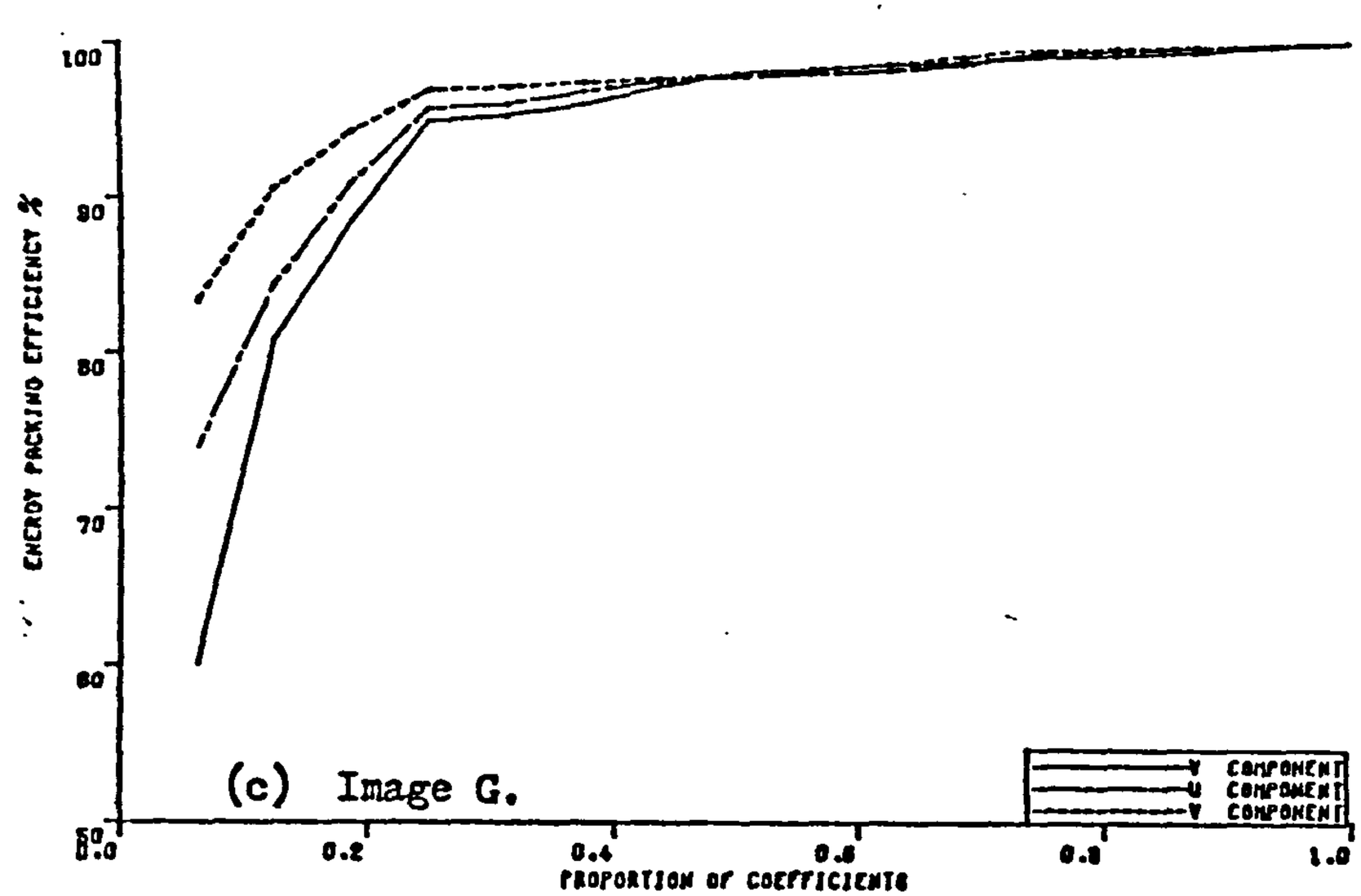


Figure (3.25). Energy packing efficiency of Components transform at $4f_{sc}$ for different images and a vector size of N=64.

3.3. General Comparison for Different Colour Signal Transforms:

Results obtained so far can be summarized as follow..

1. Generally, components transform is more efficient as far as energy compaction is concerned,(except for the case of $4f_{sc.}$), but as seen from Sections (3.1.3. and 3.2.3.) it is in fact three transforms, with all what this implies in complexity and computational efforts.

2. The laced transform, as expected, is much better than direct transform, in the case of $3f_{sc}$ in compacting the spatial energy.

3. Direct transform in case of $3f_{sc}$ is the poorest regarding energy packing efficiency. However, in the case of $4f_{sc}$, direct transform is, by far, the most efficient. As seen from Section (3.2.1.), it has the potentiality of compacting almost all the energy in three coefficients only, especially for smaller vector sizes, like (8— 32).

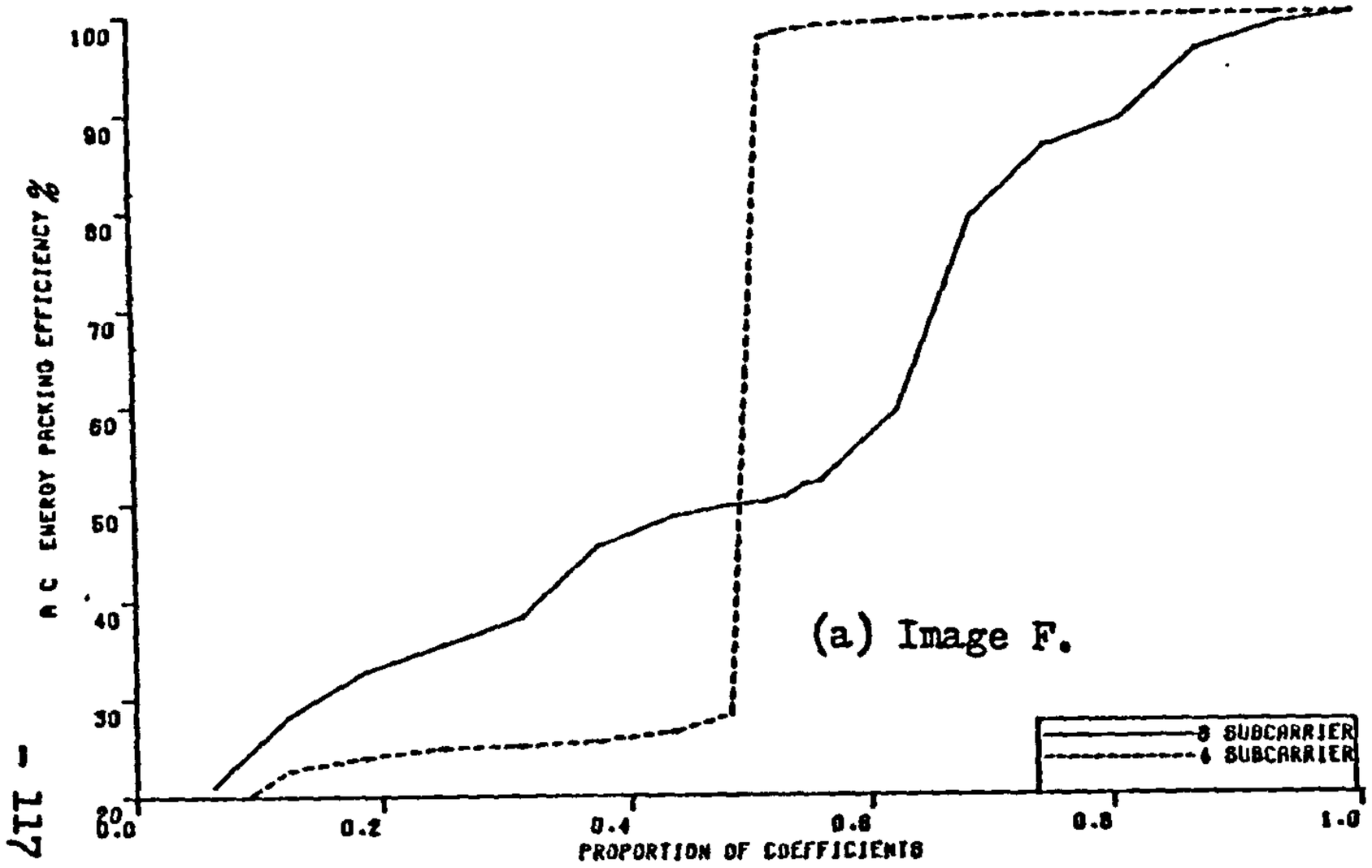
4. Except for the case of direct transform, the slight increase in efficiency by using $4f_{sc}$ sampling frequency does not seem to justify its use, as it will contribute to increasing the over-all bit rate (by increasing the number of samples and hence, number of coefficients by 33% over the $3f_{sc}$ frequency case).

Figure (3.26) compares the energy packing efficiency for the two sampling frequencies and two images in case of direct transform of vector size N= 64. Efficiency in $3f_{sc}$ case is higher in the first half of orders, while the sharp increase in efficiency of $4f_{sc}$ sampling is apparent just after the inclusion of the first term in the upper half of the spectrum. It is also clear that efficiency of $4f_{sc}$ is less dependent on image category (in the second half) than the $3f_{sc}$.

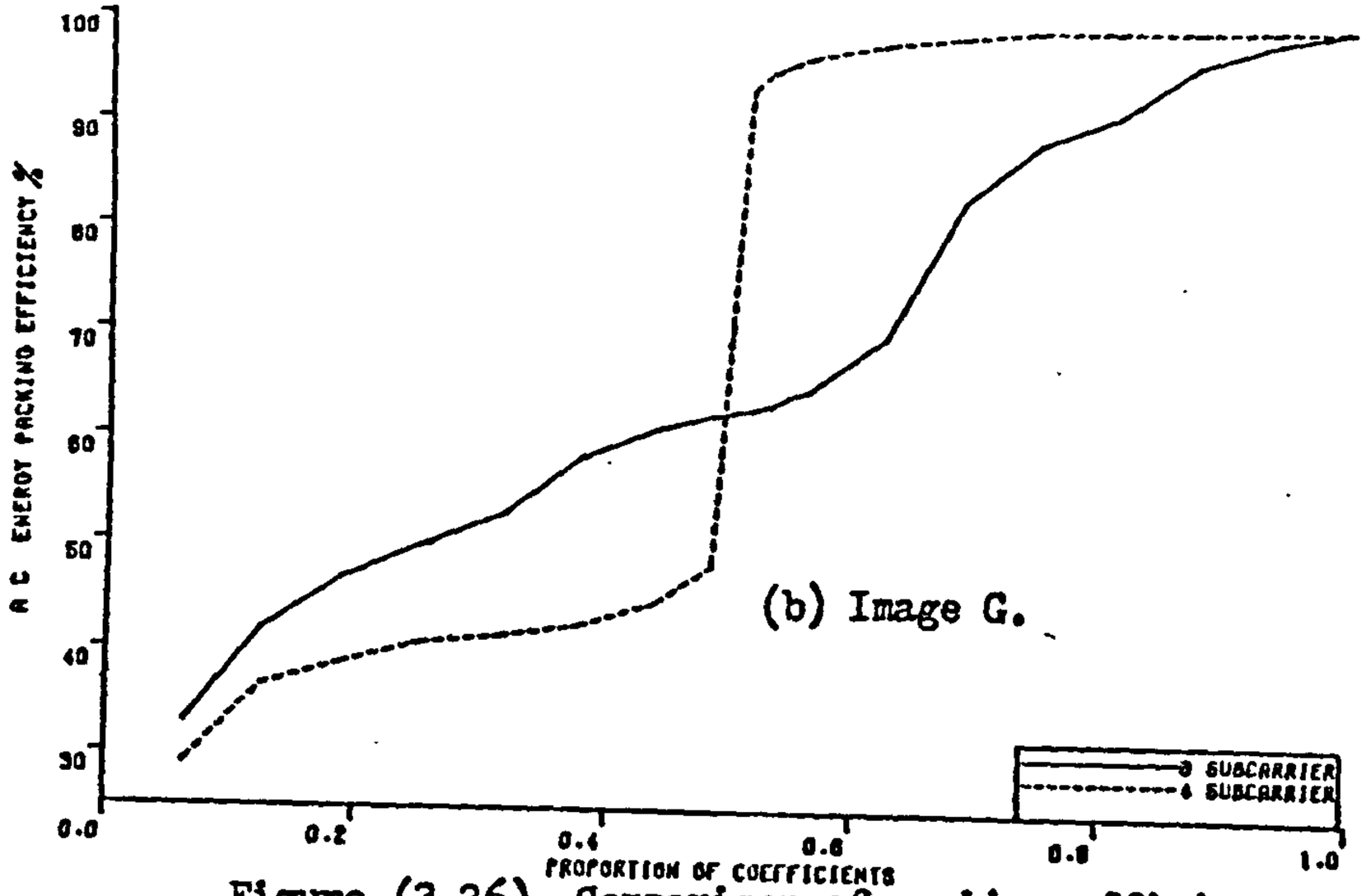
Figure (3.27) compares the two sampling frequencies in the case of laced transform of a vector size of N= 32, for two images. Slight change is noticed between the two frequencies, while noticeable difference is clear between the two images.

Figure (3.28) shows the u component and the v component transforms for two images and a vector size of N= 64. For these two component transforms, the efficiency is virtually independent of image. Again, a slight increase in the case of $4f_{sc}$ sampling is noticeable.

As a conclusion, it could be said that for a laced transform, a low sampling frequency (as $3f_{sc}$) is preferable in order to lower the number of samples. For direct transform however, the $4f_{sc}$ is much more efficient with smaller transform sizes.

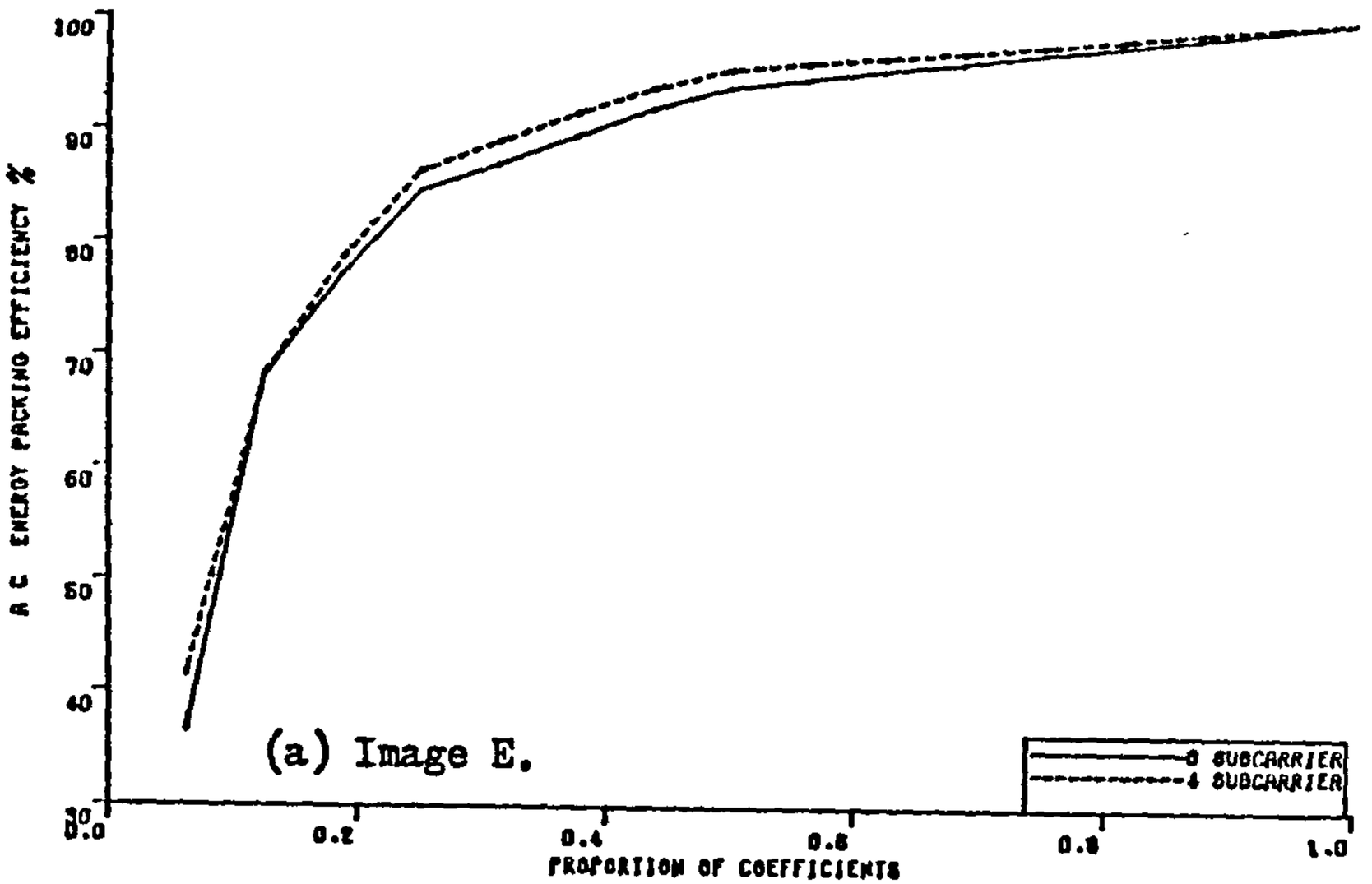


(a) Image F.

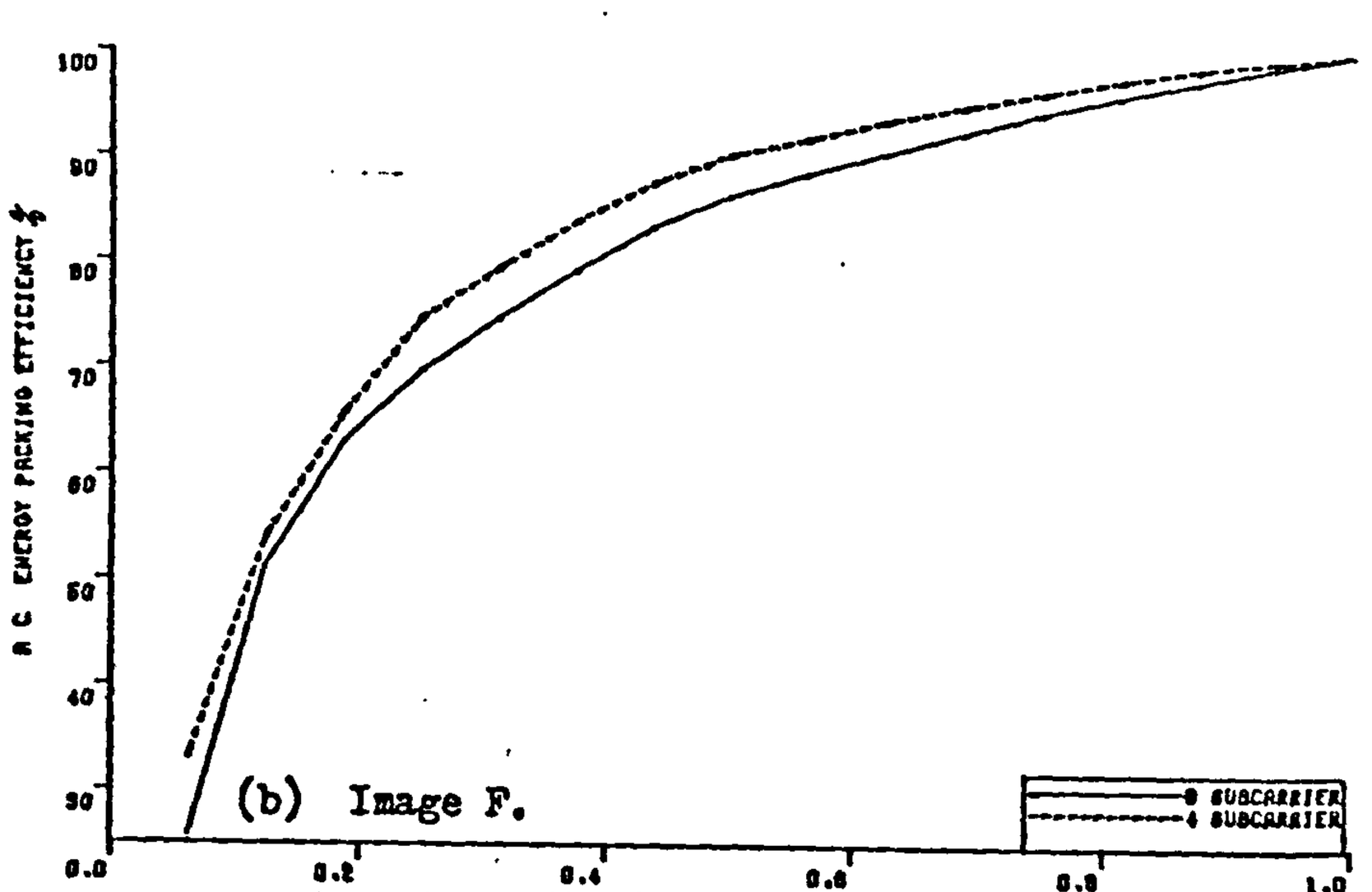


(b) Image G.

Figure (3.26). Comparison of packing efficiency for two sampling frequencies, and N=64.

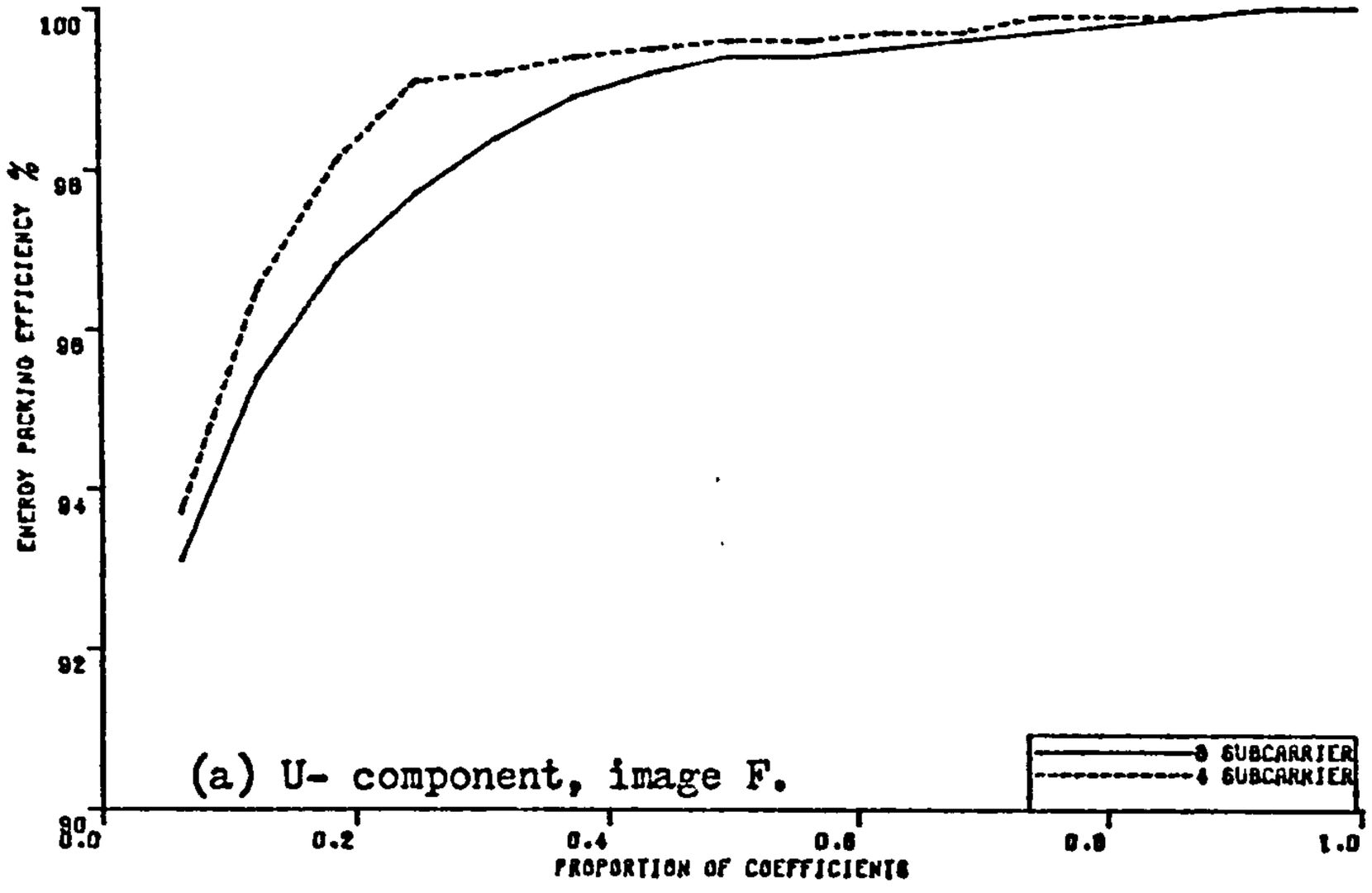


(a) Image E.

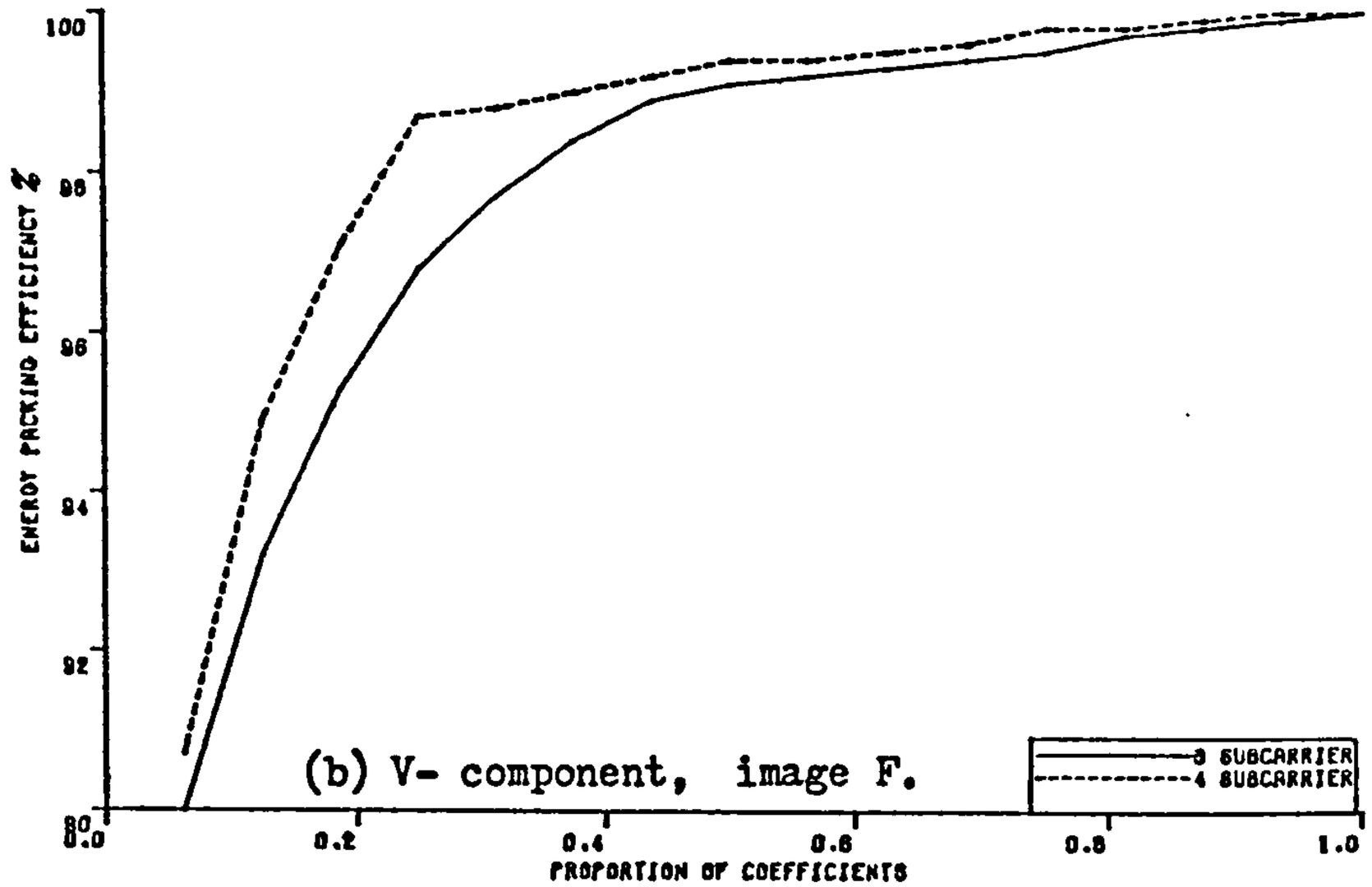


(b) Image F.

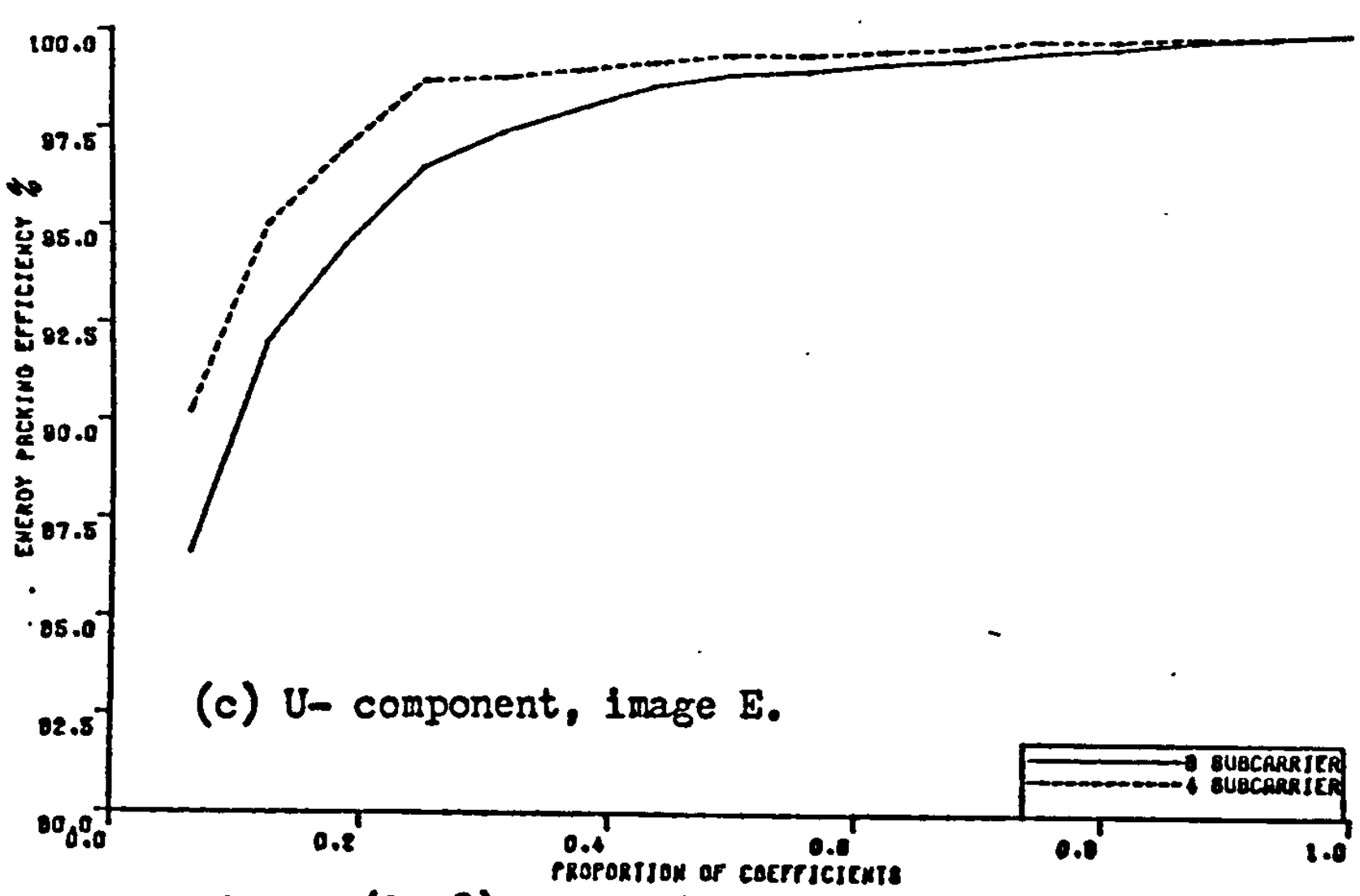
Figure (3.27). Comparison of packing efficiency for two sampling frequencies, and N=32.



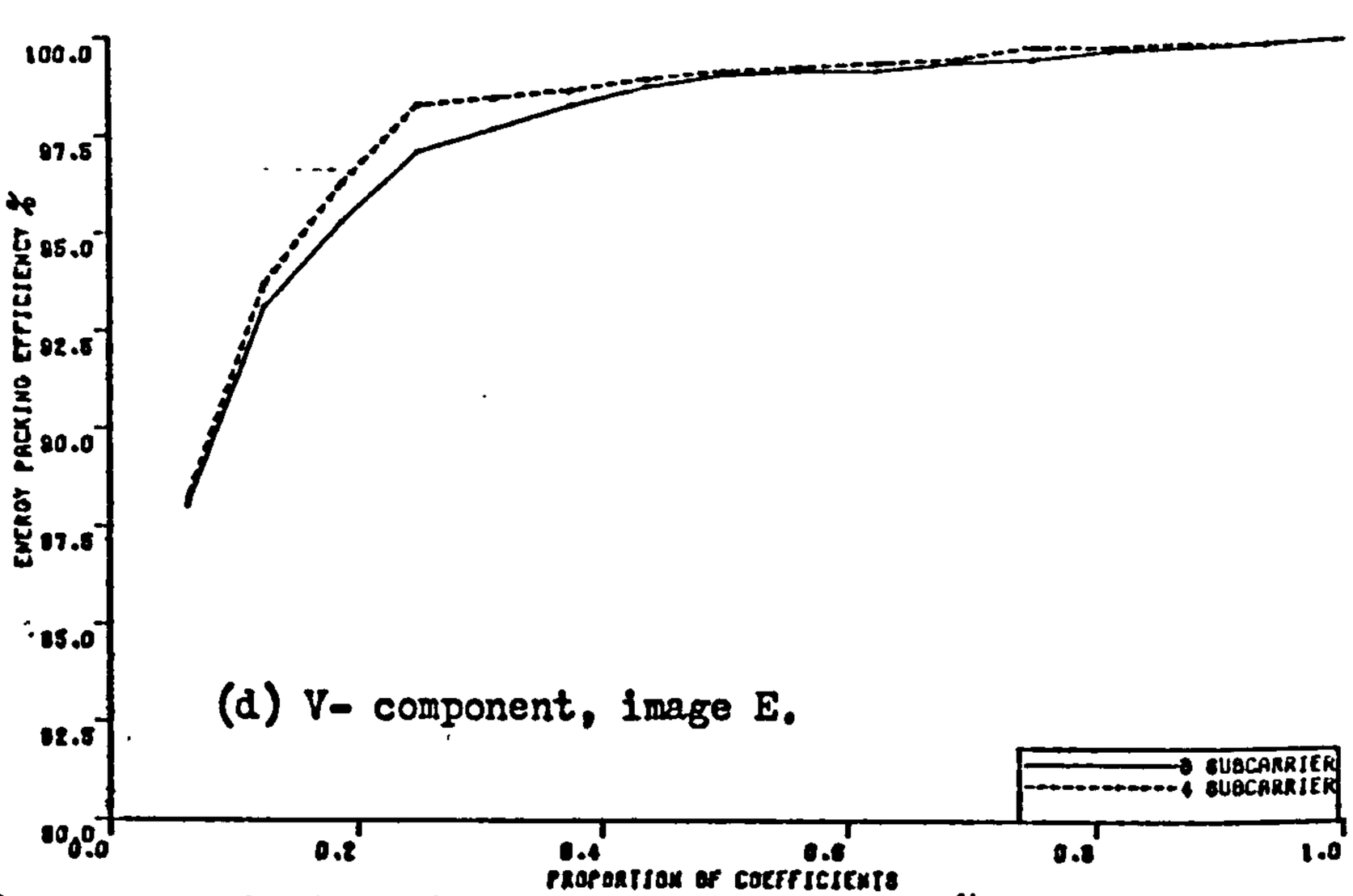
(a) U- component, image F.



(b) V- component, image F.



(c) U- component, image E.



(d) V- component, image E.

Figure (3.28). Comparison between sampling frequencies in Components transform, vector size of N=64.

CHAPTER FOUR

NORMALIZATION ERROR

Normalization error is that error which results from normalizing the values of transform coefficients to the ranges of values of the original spatial domain samples. Effects of normalizing are computed for different transform sizes in both line and block transformations. The different versions of mean-squared-error criterion, which are used for objective assessment, are defined and discussed, and some numerical values are linked with some subjective assessment tests previously mentioned in other literature. Work in this Chapter is restricted to monochrome signals only, as no great differences between monochrome and colour signals are expected in this respect.

As a result of this investigation, an upper limit on the transform size is recommended to avoid excessive distortion in the reconstructed image from the process of normalizing before any encoding or decoding is involved.

4.1. Introduction:

It has been shown in Chapters 2 and 3 that, as the transform size increases, the degree of decorrelation among transform coefficients also increases. Mean values and standard deviations of higher sequency coefficients generally decrease with transform size N. This allows efficient exploitation of redundancy in the original spatial domain by decreasing, or even neglecting completely, the role of high sequency terms in reconstructing the original data. Unfortunately, this increase in vector or block size cannot be continued without limits. The problem with such an increase is that the absolute values of coefficients also increase, and the binary words required to represent them increase in length.

From Equation (1.17), Chapter 1, in reconstructing the original data from transform coefficients, the output from an inverse transform should be "normalized" by a factor $N = 2^n$ which is the transform size. Ideally, this should be done after the final inverse transformation. But as a result of discussion in last paragraph, and especially when N is large, it may be more convenient, as far as the hardware word length is concerned, to,divide the coefficients themselves by the normalizing factor N before encoding and transmission in order to get their dynamic ranges at the same levels of magnitudes as originals. A third possibility is to divide the coefficients values by $\sqrt{N}$,and then normalizing the recovered data after inverse transformation by another factor $\sqrt{N}$. These three possibilities will be referred to here as Post-normalizing, Pre-normalizing, and Double half normalizing.,respectively.

Obviously, this normalizing process will result an error of its own, even without taking any other processing steps in the transform domain. Although this error, resulting from limiting the word length, acts as the most important constraint on the transform size, it did not, so far, receive the attention it deserves from most of researchers working in the field.

One reason for that, may be that a great proportion of experiments and studies concerning the Hadamard transform was carried out by computer simulation[50-53,69-73], which allows processing of extremely wide ranges of values for any coefficient or quantizing parameter. Another possible reason for no specific treatment of the problem of normalization error may be that the majority of researchers dealing with transformations, cosidered the total errors without discriminating among different sources of these errors.

Knab [74], studied the normalization noise (which he called round-off noise), for a particular case. He considered block transformation with a block size of 8x8 samples, and an original 6- bit PCM image. He then subjectively assessed the reconstructed image for two different cases. First, by rounding off the coefficients of line- (horizontal-, or semi-) transformation prior to column- (vertical-) transformation. Second, the rounding off was performed only after the vertical (final) transformation. He presented some results in these two cases, as well as the results of reconstruction after quantization. He came to a conclusion, that :

" an eight bit precision hardware is sufficient to keep the effects of round off error to an acceptable level when performing the two dimensional 8x8 Hadamard transform on 6-bit imagery data ". No objective assessments were given as for the rounding off effects alone, without combining the quantization effects.

In this present study, normalization error effects were investigated for different transform sizes in both the line and block transformations. The objective criteria used in judging and assessing these effects are different forms of the Signal to Noise Ratio.

4.2. Fidelity Criteria:

The ultimate receiver of TV signals is the human eye. Therefore, a suitable fidelity criterion should be a subjective assessment of the TV picture in question. However, for theoretical investigations, and where the necessary means of subjective assessment are not available, the need arises for a computable objective assessment criterion. The one which is widely accepted is the Mean Square Error. Although it has no sufficient theoretical or practical justifications for its use over other distortion criteria,(75), it is still the most commonly used. Other criteria are being widely developed for use in conjunction with image prcessing(75-78)

Objective criteria used here are different versions of normalized mean square error. Because there are no general agreements on the act of normalizing the MSE, the following names will be used here as defined below.

4.2.1. Over all Normalized Mean Square Error ONMSE :(79)

This is a measure of the normalized mean square error (difference), between a single frame data $f_{(x,y)}$, and the reconstructed image points $\hat{f}_{(x,y)}$, where x and y are the spatial coordinates. Normalization is achieved by dividing the mean square error by the mean signal energy all over the frame area, and hence the name ' over all normalized'. Expression for the ONMSE is as follows:

$$ONMSE = \frac{1/M \sum_{x=1}^{Nx} \sum_{y=1}^{Ny} \left[f_{(x,y)} - \hat{f}_{(x,y)} \right]^2}{1/M \sum_{x=1}^{Nx} \sum_{y=1}^{Ny} \left[f_{(x,y)} \right]^2} \quad \ldots\ldots \quad (4.1).$$

where Nx =No. of samples along x direction,

Ny =No. of samples along y direction,

M =total number of samples =Nx.Ny

In the special case where Nx=Ny =$\sqrt{N}$, M=N.

An over all normalized signal to noise ratio can be expressed in the usual form as :

$$ONSNR = -10 \log_{10} ONMSE \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.2.)$$

4.2.2. Peak-to-Peak Signal-to-Root Mean Square Noise Ratio PPSNR :(79,80)

This error is calculated by-normalizing the mean square error all over the image area to the(maximum intensity in the image $Y_{max.}$)^2

An expression for this criterion is:

$$PPMSE = \frac{1/M \sum_{x=1}^{Nx} \sum_{y=1}^{Ny} \left[f_{(x,y)} - \hat{f}_{(x,y)} \right]^2}{Y^2_{maximum}} \quad \ldots\ldots\ldots\ldots\ldots\ldots (4.3.)$$

where $Y_{maximum}$ =maximum luminance value of $f_{(x,y)}$.

Again, a signal to noise ratio is coputed as:

$$PPSNR = -10 \log_{10} PPMSE \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.4.)$$

The normalization is performed relative to the maximum luminance value to exploit one of the human eye characteristics. This is the inaccuracy of the eye in assessing the noise at the highly luminant regions(81) . However, this SNR measure is somewhat optimistic as it does not take into account the relatively higher effects of noise at the poorer luminant areas.

4.2.3. Locally Normalized Signal to Noise Ratio LNSNR :

This is a rather conservative, or even pessimistic, MSE criterion. It differs from the PPMSE above, in that the normalization is done for each pixel separately by its corresponding luminance value. Thus, a relation for such an error is:

$$LNMSE = 1/M \left[\sum_{x=1}^{Nx} \sum_{y=1}^{Ny} \left[\frac{f_{(x,y)} - \hat{f}_{(x,y)}}{f_{(x,y)}} \right]^2 \right] \quad \ldots\ldots\ldots (4.5.)$$

Also, a locally normalized signal to noise ratio will be defined as:

$$LNSNR = -10 \log_{10} LNMSE \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.6.)$$

4.2.4. Objective and Subjective Criteria Relations:

For a comparative assessment of objective measures mentioned so far some results are recalled where subjective assessments were also given. Among the previously mentioned objective criteria, the PPSNR was the one mostly used. In a work by Chen and Smith[31], a subjective assessment for a 37 dB PPSNR colour image seemed to be excellent compared with the original. For monochrome images, a comparable subjective assessment seemed applicable for a (30 - 32) dB PPSNR levels. In another work by Ishii[80], a PPSNR of between (27.8 - 28.9) dB seemed acceptable for a variety of monochrome images. In the same range of 28 dB, another work[82] was reported, although, unfortunately, the specific version of objective SNR was not mentioned.

4.3. Normalizing and Rounding off:

The study of normalizing and rounding off effects was performed by simulating both the transformation and normalizing on a digital computer, (ICL 1904S). All arithmetic operations of transformationfrom the spatial domain were allowed to take any values without constraints. Then, simulating the hardware word length limit before transmission, transform coefficients were normalized to the same levels of magnitudes as the orignal samples values, by applying the following process

$$H_i \text{ (normalized and rounded)} = \frac{H_i \pm N/2}{N} \quad \ldots\ldots\ldots (4.7.)$$

where H_i = the value of i^{th} coefficient,

N = transform size = 2^n.

The ($\pm$) signs are used with H_i positive or negative respectively.

Division processes were carried out in integer arithmetic to simulate the limiting and round off. Normalized and rounded coefficientswere then inverse transformed and compared with original data. Errors calculated after that were,obviously, due to normalizing and rounding off.

Normalizing was performed, in all the simulations shown below, in the three different ways mentioned before, namely, Postnormalizing, Prenormalizing, and Double half-normalizing. No appreciable differences in results were observed. Results shown here are for the Prenormalizing, where full normalization is simulated before encoding.

4.3.1. Normalization Error in Line Transform:

Simulation results for vector sizes ranging from N= 4 up to N= 128 (n= 2 - 7), for the four image categories are shown in Table (4.1). As a means of comparison, and for a quick assessment,the objective measures of different degrees of bit reduction in original data of one image (A), are shown in Table (4.2.). The table shows the same objective criteria for image (A), after being subjected to a decrease in its representative word length by 1,2,3,4,and 5 bits. Thus, the table shows the corresponding SNR's for the cases of 7,6,5,4,and 3 bits word long PCM image, respectively, obtained from the original 8-bit image.

Results of Tables (4.1. and 4.2.) are plotted in Figure (4.1.), using only the PPSNR criterion.

From Tables (4.1. &4.2.) and Figure (4.1), it is seen that, on average, and nearly for all images tested, the decrease in SNR is nearly 2.8 dB for each bit increase in n (i.e. for each doubling in the transform size N).

Table (4.1.). Normalization SNR for different vector sizes and image categories.

Image category	Vector size.	ONSNR in dB.	PPSNR in dB.	LNSNR in dB.
A	4	49.89	52.67	48.59
B	4	48.90	52.54	47.98
C	4	49.90	52.79	48.59
D	4	47.74	52.54	45.59
A	8	47.37	50.16	45.94
B	8	46.39	50.04	45.43
C	8	47.58	50.46	46.13
D	8	45.09	49.89	42.98
A	16	44.68	47.47	43.15
B	16	43.81	47.45	42.83
C	16	44.91	47.80	43.33
D	16	42.34	47.14	40.09
A	32	41.84	44.62	40.20
B	32	41.28	44.92	40.17
C	32	41.44	44.33	40.07
D	32	39.34	44.14	37.15
A	64	38.90	41.69	37.19
B	64	38.52	42.17	37.25
C	64	38.29	41.18	36.78
D	64	36.30	41.10	34.24
A	128	35.91	38.70	34.21
B	128	35.85	39.50	34.58
C	128	35.15	38.03	33.68
D	128	33.24	38.04	31.13

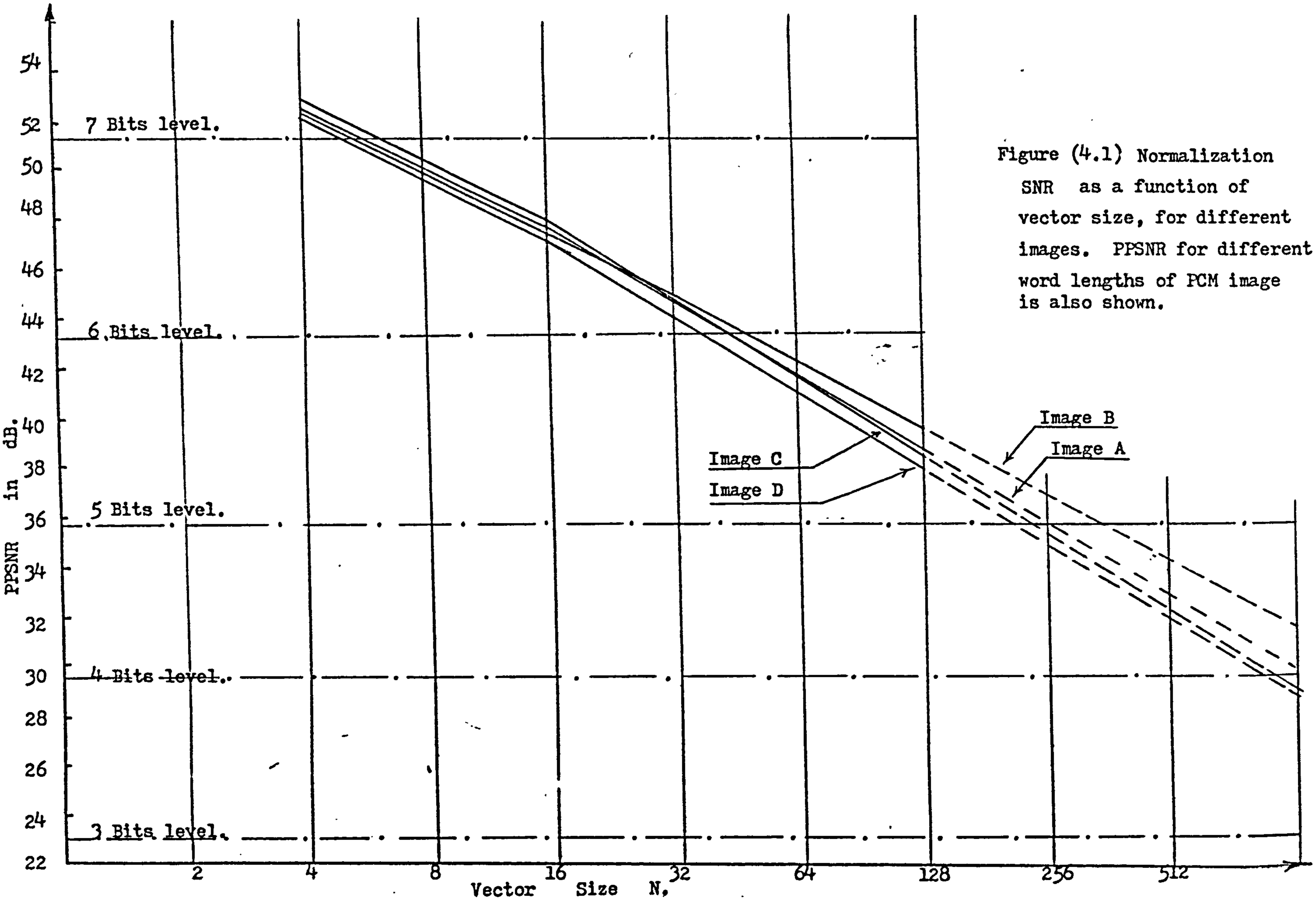


Figure (4.1) Normalization SNR as a function of vector size, for different images. PPSNR for different word lengths of PCM image is also shown.

By comparison from Figure (4.1), it is clear that a normalized 4 pixel vector will result in a picture which is better than a 7 bit PCM image. For vector sizes of 8, 16, and 32 pixels, the resulted images, after normalizing and rounding off, are better than a 6 bit PCM image. When the vector size increases to 64 pixels, the image quality is just comparable with, or slightly inferior to, that of a 6 bit PCM image. This means that, for a word lenght of 8 bits, a transform size should not, in general, exceed 64 pixels.

Table (4.2). SNR's due to bit reduction of original 8- bit PCM image. (Image A).

Reduction to	ONSNR dB.	PPSNR dB.	LNSNR dB.
7 bits	48.56	51.34	47.31
6 bits	40.58	43.37	39.33
5 bits	32.98	35.77	31.80
4 bits	26.81	29.60	25.40
3 bits	20.26	23.05	19.01

4.3.2. Normalization Error in Block Transform :

The work reported in previous Section for line transform was extended to block transformation as well. In all work reported so far in literature concerning quantization in transform domain[50-52,76-78,83,84] blocks were considered only in the square form. No attempt has been reported so as to using non-square blocks. In this work, non-square, as well as square blocks are considered. Furthermore, different combinations of dimensions were also tried for both axes.

Again, as before, in simulation, all coefficients were allowed to take on any values during the transformation. Normalizing was then applied after the final (vertical) semi-transformation. Reconstruction by inverse transforming restored the 'approximate' original data ,and hence normalization error was calculated. Results of simulations for three different images are shown in Tables (4.3. — 4.5.). Because of the large amount of informations in this case, results for each image were grouped alone. In each table, different criteria are shown grouped with respect to the vertical dimension Ny which is the number of TV lines involved in composing the block, while the horizontal dimension Nx, is shown as a parameter. All possible likely combinations were considered from 2x2 up to 16x16. These cover almost all practical cases one can think about (or perhaps more). Neverthless, in plotting these results, it was found easier for categories comparison, to plot different images with same block dimensions together. So, one figure is devoted to each value of vertical dimension, with the horizontal dimension (vector size), and the image category as parameters. Figures (4.2. — 4.5.) show these results.

Table (4.3.). Normalization SNR for different block sizes and shapes. Image A.

Block size. Lines Ny	Vector Nx	ONSNR in dB.	PPSNR in dB.	LNSNR in dB.
2	2	49.83	52.62	48.56
2	4	47.39	50.18	45.95
2	8	44.74	47.53	43.13
2	16	41.99	44.78	40.31
4	2	47.28	50.07	45.88
4	4	44.69	47.48	43.14
4	8	42.03	44.81	40.31
4	16	39.25	42.03	37.46
8	2	44.55	47.33	43.06
8	4	41.90	44.68	40.22
8	8	39.20	41.99	37.44
8	16	36.54	39.33	34.70
16	2	41.67	44.45	40.10
16	4	39.04	41.83	37.31
16	8	36.41	39.19	34.60
16	16	33.93	36.72	32.07

Table (4.4.). Normalization SNR for different block sizes and shapes. Image B.

Block size Line Ny	Vector Nx	ONSNR in dB.	PPSNR in dB.	LNSNR in dB.
2	2	48.88	52.53	47.96
2	4	46.46	50.11	45.55
2	8	43.94	47.58	42.98
2	16	41.54	45.18	40.48
4	2	46.35	50.00	45.41
4	4	43.92	47.57	42.96
4	8	41.63	45.27	40.60
4	16	39.41	43.06	38.24
8	2	43.78	47.43	42.82
8	4	41.54	45.10	40.49
8	8	39.36	43.00	38.19
8	16	37.27	40.92	36.01
16	2	41.27	47.57	40.16
16	4	43.92	44.92	42.96
16	8	37.13	40.78	35.90
16	16	35.16	38.81	33.84

Table (4.5.). Normalization SNR for different block sizes and shapes. Image C.

Block size Line Ny	Vector Nx	ONSNR in dB.	PPSNR in dB.	LNSNR in dB.
2	2	49.85	52.74	48.58
2	4	47.68	50.57	46.18
2	8	45.23	48.11	43.65
2	16	42.37	45.25	40.69
4	2	47.55	50.44	46.08
4	4	45.39	48.28	43.79
4	8	42.92	45.83	41.41
4	16	39.97	42.86	38.31
8	2	45.16	48.05	43.59
8	4	43.08	45.96	41.46
8	8	40.63	43.51	39.04
8	16	37.47	40.36	35.75
16	2	42.59	45.47	41.03
16	4	40.51	43.40	38.93
16	8	38.07	40.95	36.49
16	16	34.96	37.85	33.18

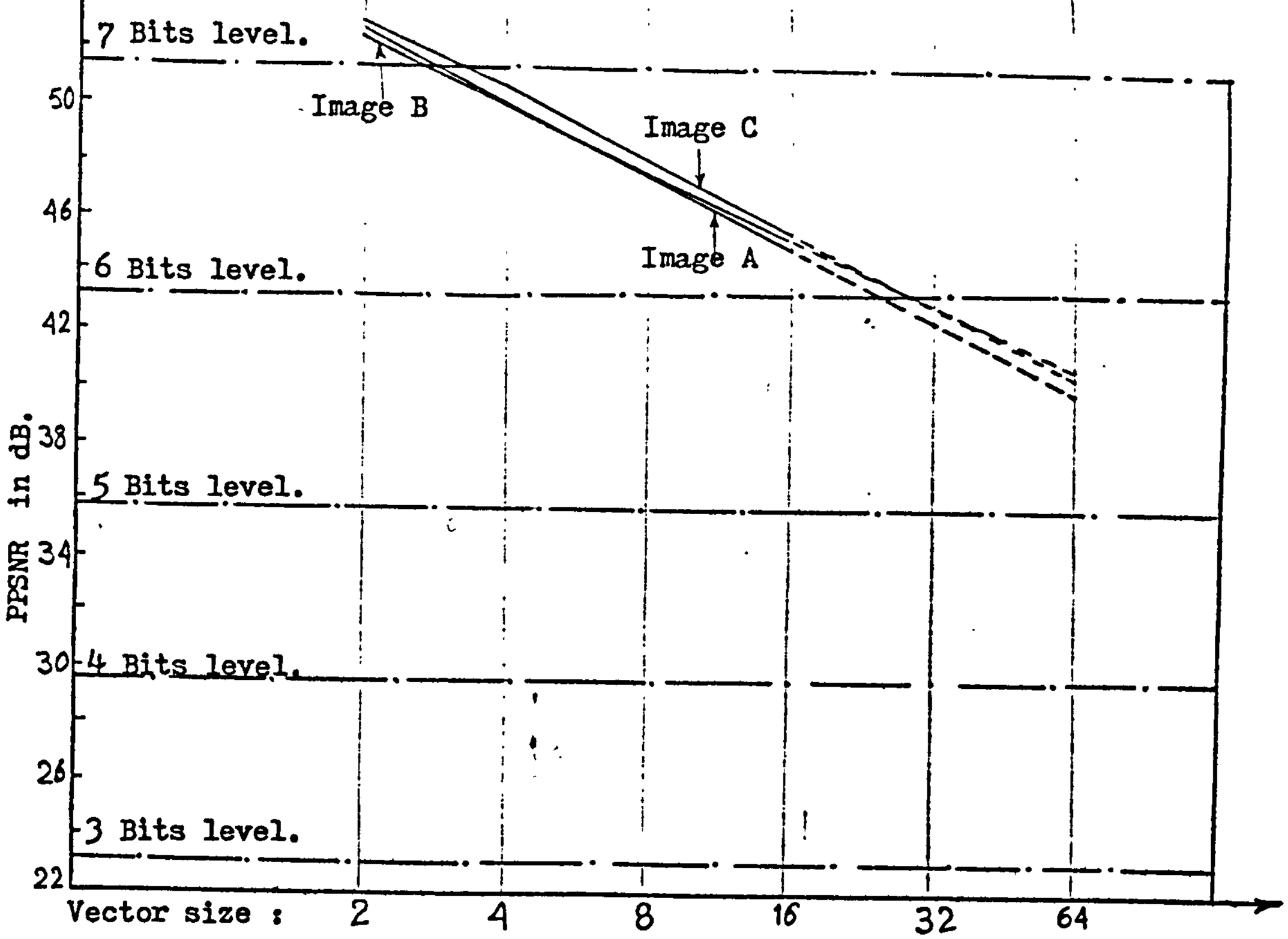


Figure (4.2). Normalization SNR as a function of horizontal vector size. Vertcal dimension is constant at 2 (number of TV lines).

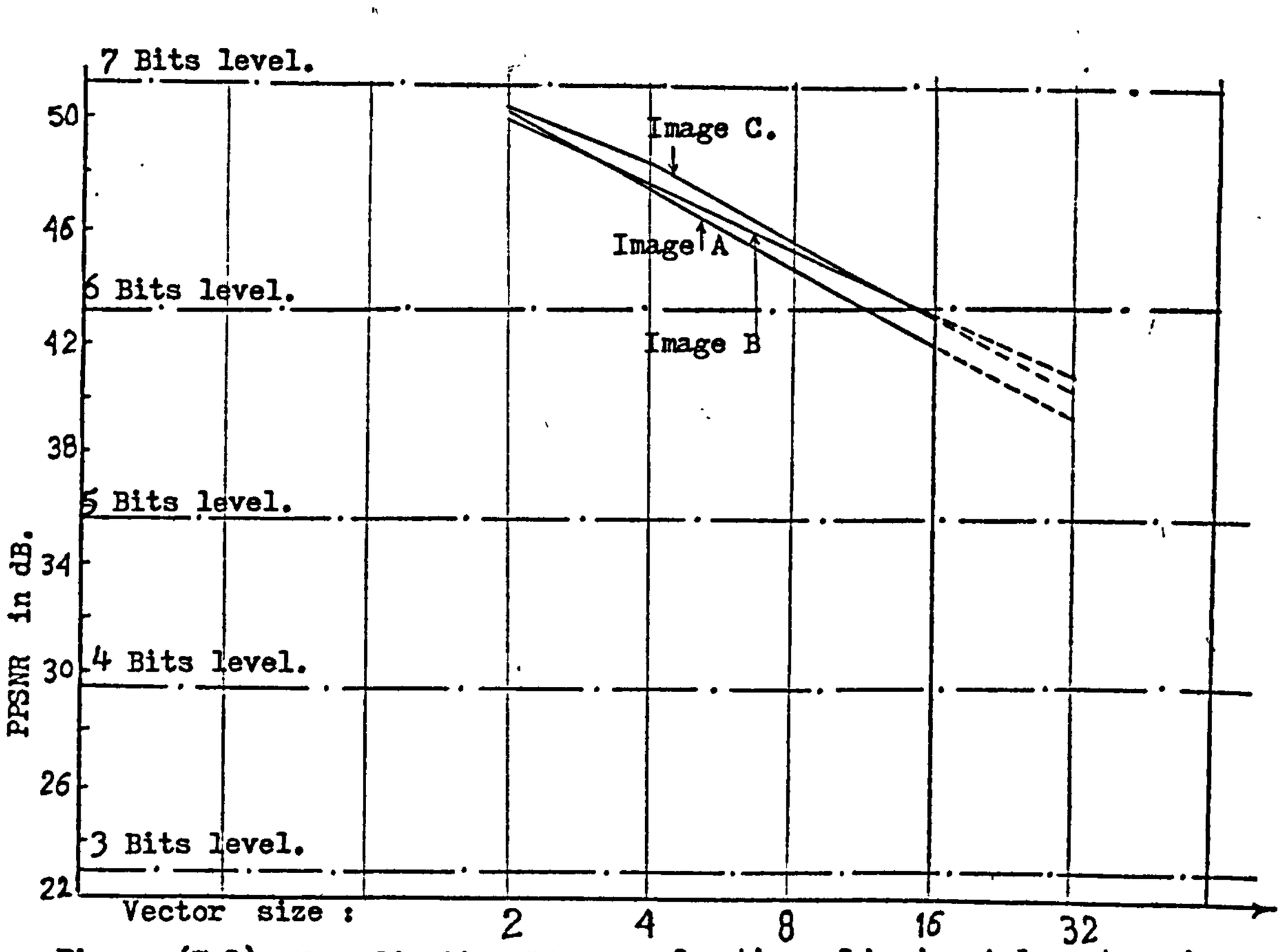


Figure (7.3). Normalization SNR as a function of horizontal vector size. Vertical dimension is constant at 4 (number of TV lines).

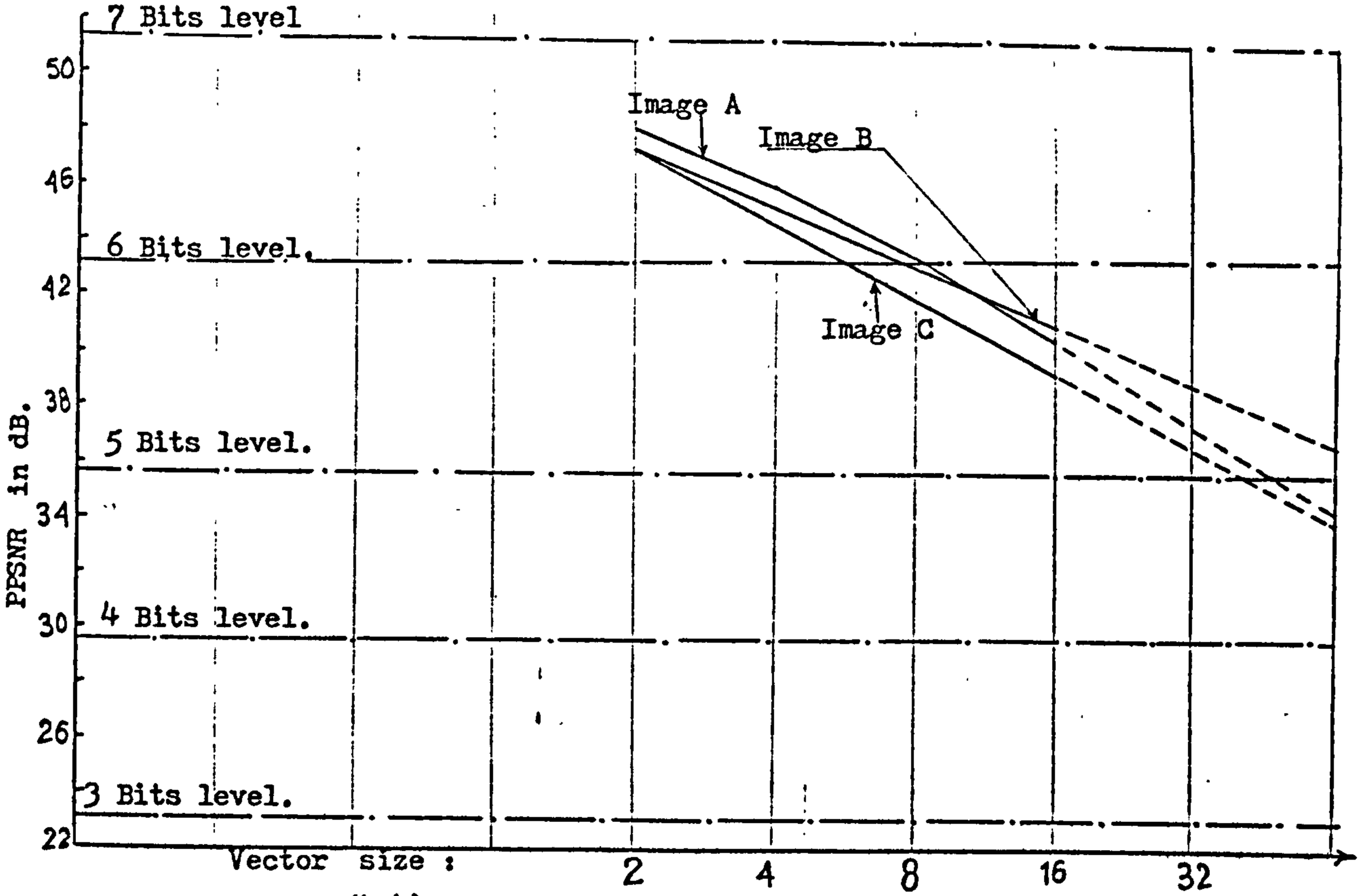


Figure (4.4). Normalization SNR as a function of horizontal vector size. Vertical dimension is constant at 8 (number of TV lines).

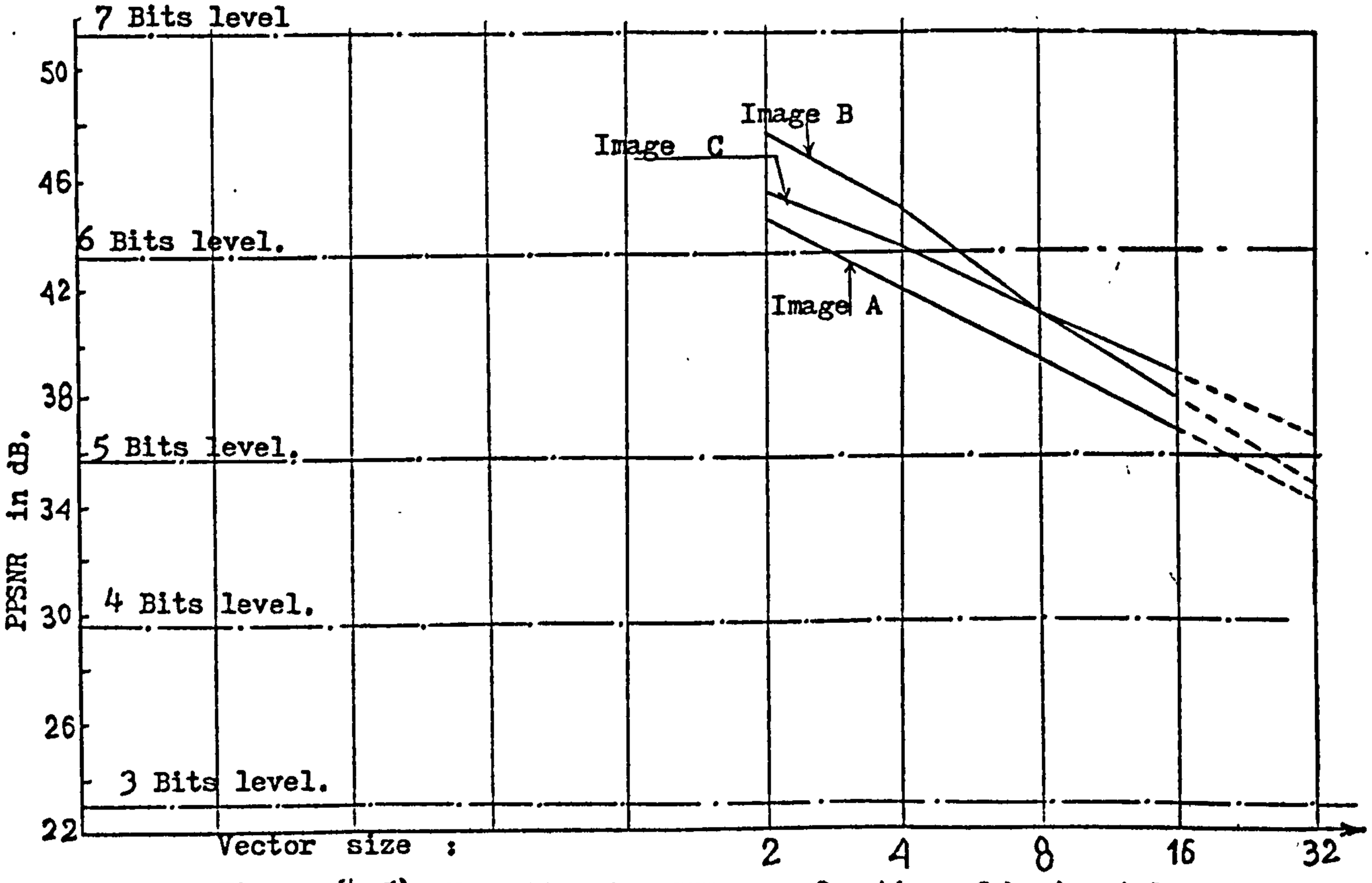


Figure (4.5). Normalization SNR as a function of horizontal vector size. Vertical dimension is constant at 16 (number of TV lines).

From Tables and Figures, some conclusions can be summarised here.

1. Generally, the ranges of values of SNR for equivalent transform sizes (i.e. N= Nx . Ny), are nearly the same as in the line transform.

2. There are slight differences between reversed dimensional blocks, of the same total sizes. For example, for image A, a block of Ny = 2 and Nx= 16 is better than Ny= 16 and Nx= 2 by about 0.33 dB. On the other hand, for image C, an 8x16 block is better than a 16x8 block by about 0.59 dB. This result may support the idea that the errors are functions of image areas.

A proposed idea to better investigate the block transformation, is to relate both dimensions Nx and Ny with correlation coefficients in corresponding directions for the local part of image considered. However, such a scheme will have to be bounded as far as the vertical dimension is concerned due to the line storages required, and because the stored lines can not be changed in number as we proceed in the horizontal direction. Due to that, it is felt that, if a block transformation is to be used, a rectangular block is preferrable, with Ny as minimum as 2, or 4, as this will greatly reduce the storage required, especially at the rec iving end.

CHAPTER FIVE

ADAPTIVE CODING OF MONOCHROME SIGNALS

For a multicoder scheme, in which one coder is used for a particular 'class' of subpictures, a 'classifying' technique is essential to identify the class and to select the appropriate coder-decoder. The existing proposed activity index is reviewed, and it was found that another objective means is necessary.

The interrelationships among different groups of coefficients in the transform are used to compute an over all ' Directing Index ', which will direct the subpicture concerned to its appropriate coder-decoder. Energy spectra of transform domain groups of coefficients are computed, and different directing indexes are considered to pick those areas which have much energy. Coders to be used with each index were also devised and tested, subject to an arbitrary bit rate of 2 bits per pixel.

Assessments of the scheme are presented. It is shown that the gain over non-adaptive coding is considerable, while the decrease than a 'hypothetically ideal' directing process is very low.

5.1. Introduction:

Several attempts have been tried to adapt thescheme of quantization to the picture characteristics. Tasto and Wintz[83], used local statistical properties of picture to divide it into a number of segments, and chose the coding scheme-suited for each segment. In a detailed work[61], they divided the blocks (subpictures) into one of three categories:

Category I : blocks containing a lot of detail.

Category II : blocks containing little detail and darker than average.

Category III: blocks containing little detail and lighter than average.

This division of images depends upon spatial activity, although not explicitly. They then developed quantizers for each category by trial and error.

The problem of objective measures of activity, detail, or busyness, of an image has not received sufficient attention so far, apart from a proposed 'activity index' [62].

5.2. Activity Index:

Gimlett[62] has proposed an activity index for subpictures to help assigning appropriate number of bits for different coefficients in the transform domain. He suggested that " an activity index A defined by

$$A = \sum_{i=2}^{N} a_i |F_i| \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (5.1)$$

$$\text{or} \qquad A = \sum_{i=2}^{N} a_i F_i^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (5.2)$$

could well serve as an objective measure of the busyness of a subpicture of N samples. The F_i are the transform coefficients, and the a_i, weighting factors, perhaps inversely proportional to the variances of F_i. The summation omits the first coefficient (assumed to be the dc term), because it is normally the only coefficient with non-zero mean. In this way, a subpicture with no detail would have A= 0'".

Figure (5.1.a) shows the probability density , and probability cumulative functions of the activity index as proposed by Gimlett. This was computed with a_i set to unity and for a particular image quantized to 8-bits. The original was a 128x128 pel image, divided into 256, (8x8) subpictures.

He then suggested grouping the subpictures into different categories on the bases of their activity indexes, and encoding each category differently. Four categories (I, II, III, and IV in the Figure) are obtained by using the 25%, 50%, and 75% ordinates of the cumulative probability curve to separate the categories. In this way, the four classes are of equal numbers of subpictures. Applying combination domain sampling and thresholding procedure[85], an example has been shown, for which Figure 5.1.b summarizes the number of bits allocated for each of the coded coefficients for the (8x8) subpictures image. The non-zero numbers are the number of bits transmitted for the corresponding coefficient. Dashed lines segment the array into the different zones where adapting thresholding is to be applied. The number 0 in these zones indicates that the largest coefficient in this zone plus its address are to be coded and transmitted.

From the figure, the average number of bits/subpicture can be calculated as shown in Table (5.1).

Table (5.1). Total number of bits for each class.

Class.	I	II	III	IV
Bits for adaptively selected coefficients.	6	12	22	36
Bits always transmitted.	28	34	40	70
Class code bits.	2	2	2	2
Total bits.	36	48	64	108

Assuming equal distribution of the four classes, the average bit rate will then be:

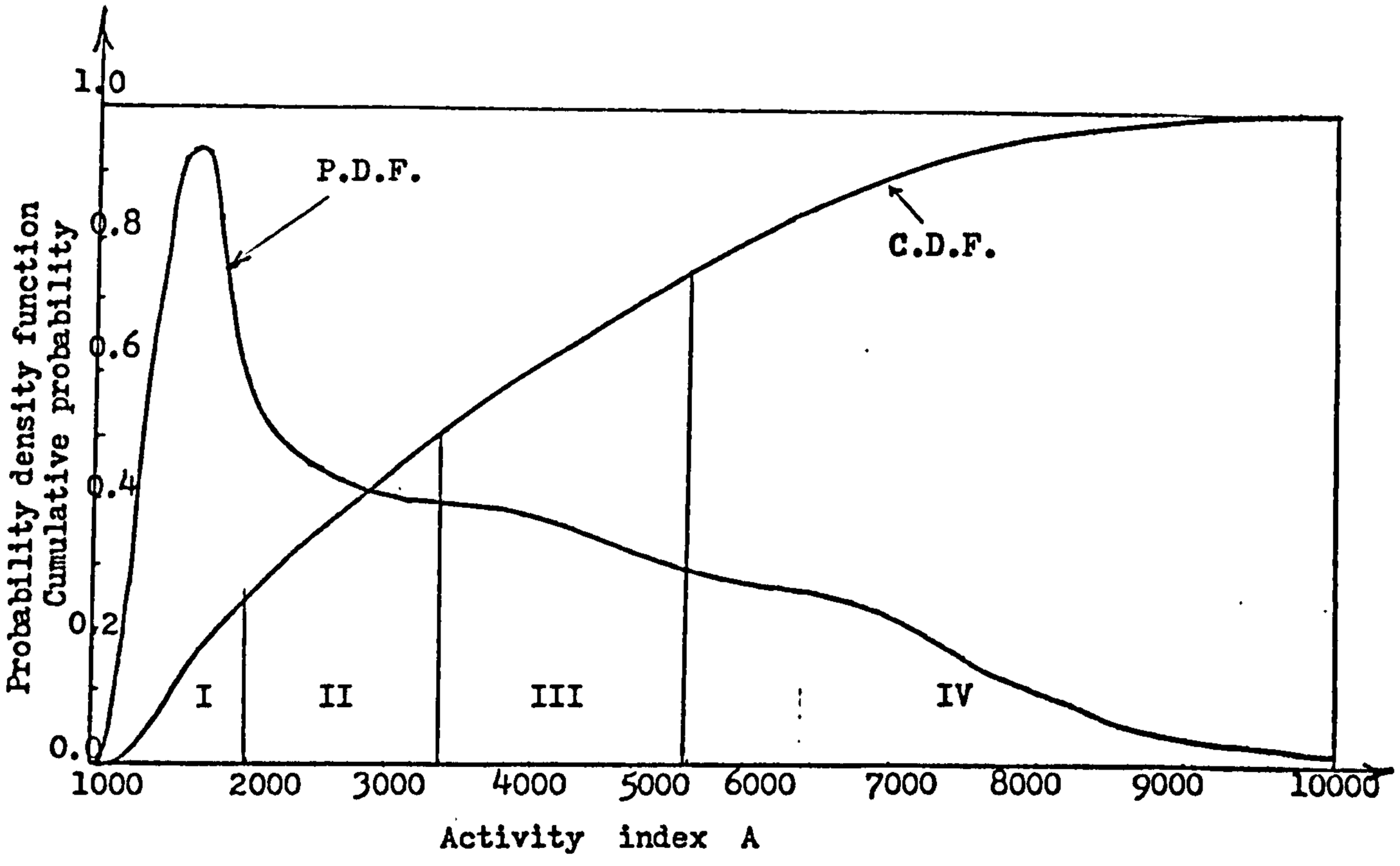


Figure (5.1.a). Probability functions of Activity index A.

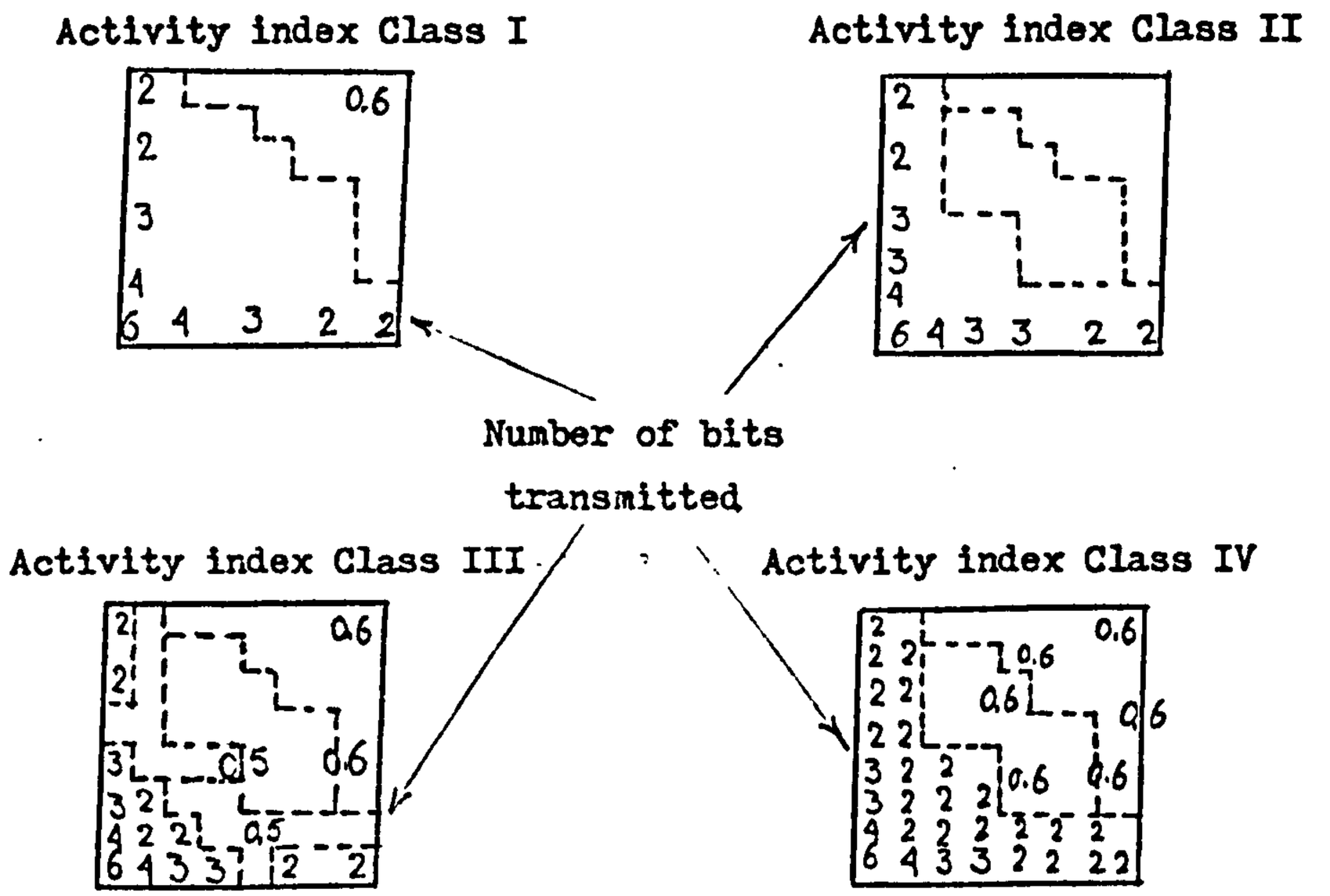


Figure (5.1.b). Bit allocation for the four activity classes.

Bit rate $= \sum_{j=1}^{4} P_j . b_j = \frac{1}{4}$ (36+ 48+ 64+ 108)= 64 bits/subpicture.
This will lead to an average bit rate of 1 bit/pixel.

However, it must be pointed out here, that this was an entirely illustrative example and no attempt has been made to minimize mean square error or any subjective error criterion. The scheme has not been even tried(62). It should also be noted that the scheme proposed in that example assumes a variable length word coding (ranging from 36 to 108 bits/subpicture as seen from the above table).

The activity index proposed by Gimlett, was tested during the course of this study as an objective way for adaptive coding. The first difficulty met in obtaining such an index was the lack of information in connection of the weighing factors a_i in the Equation (5.1.). The suggestion that they may be inversely proportional to the variances of the transform coefficients, implies that the coefficients have to undergo statistical analyses before the index could be calculated. It is not even clear if these analyses will be done only once as a general model of TV pictures, (which proved in Chapters 2 and 3 to be far from possible), or should they be done for each subpicture alone. Setting these factors to ones, as Gimlett himself has done, an attempt has been made here to get through. Figure(5.2.) shows the cummulative probability functions for the activity index in the four images concerned. The vector size is N= 16, and the coefficients are normalized to the same ranges of magnitudes as the original samples. Figure (5.3.) shows the probability for a hypothetical image composed of the previous four. The figures show clearly that the variations among the images are too wide to agree with a proper model for them all. Maximum values of activity index varies from as low as 78, (on a scale of 255 for original sample values), to as high as 153, which is nearly the double. Applying the same class dividing rule as used by Gimlett, the four equally distributed ranges of activity index values are

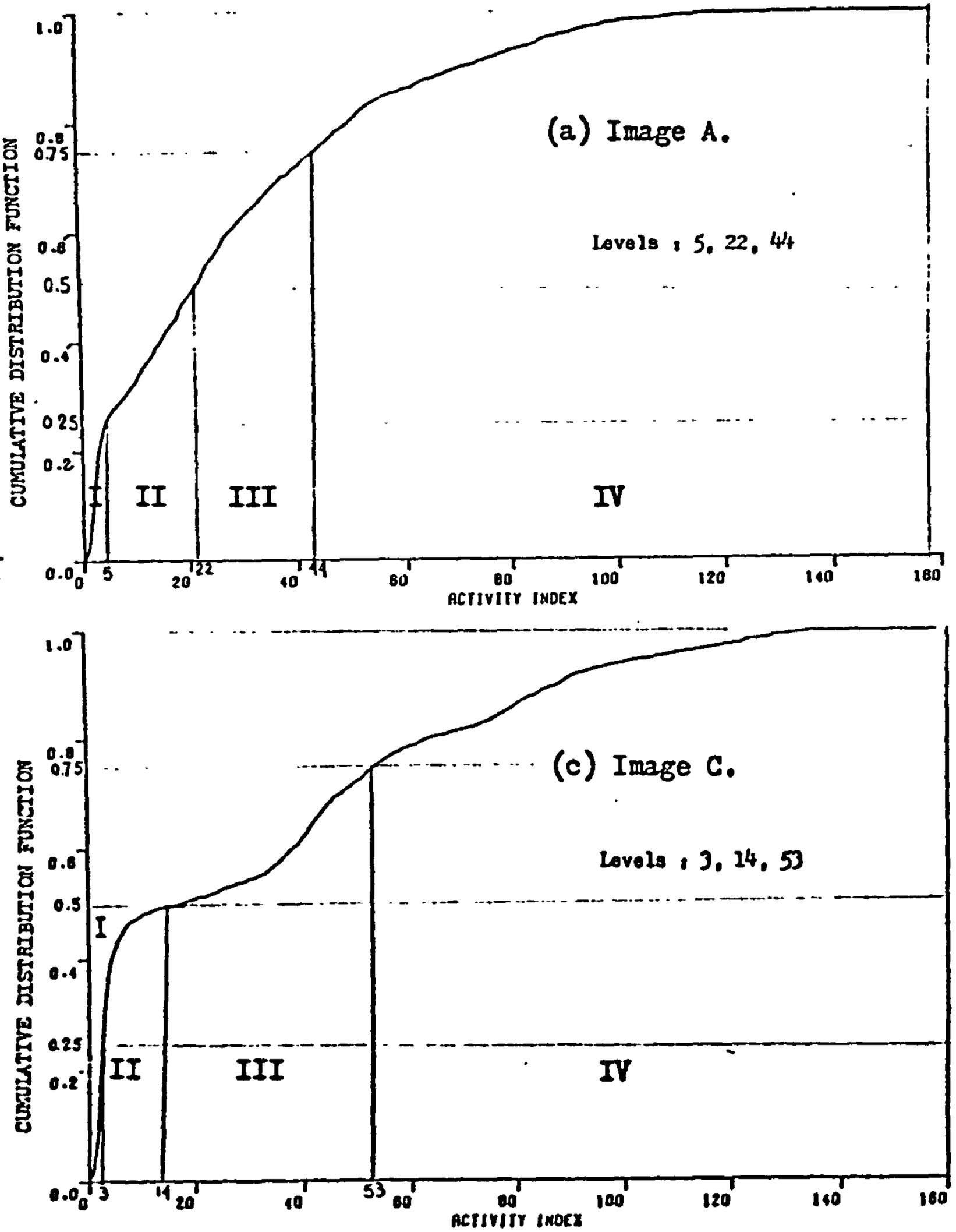


Figure (5.2). Four activity classes of different images, using the Activity Index.

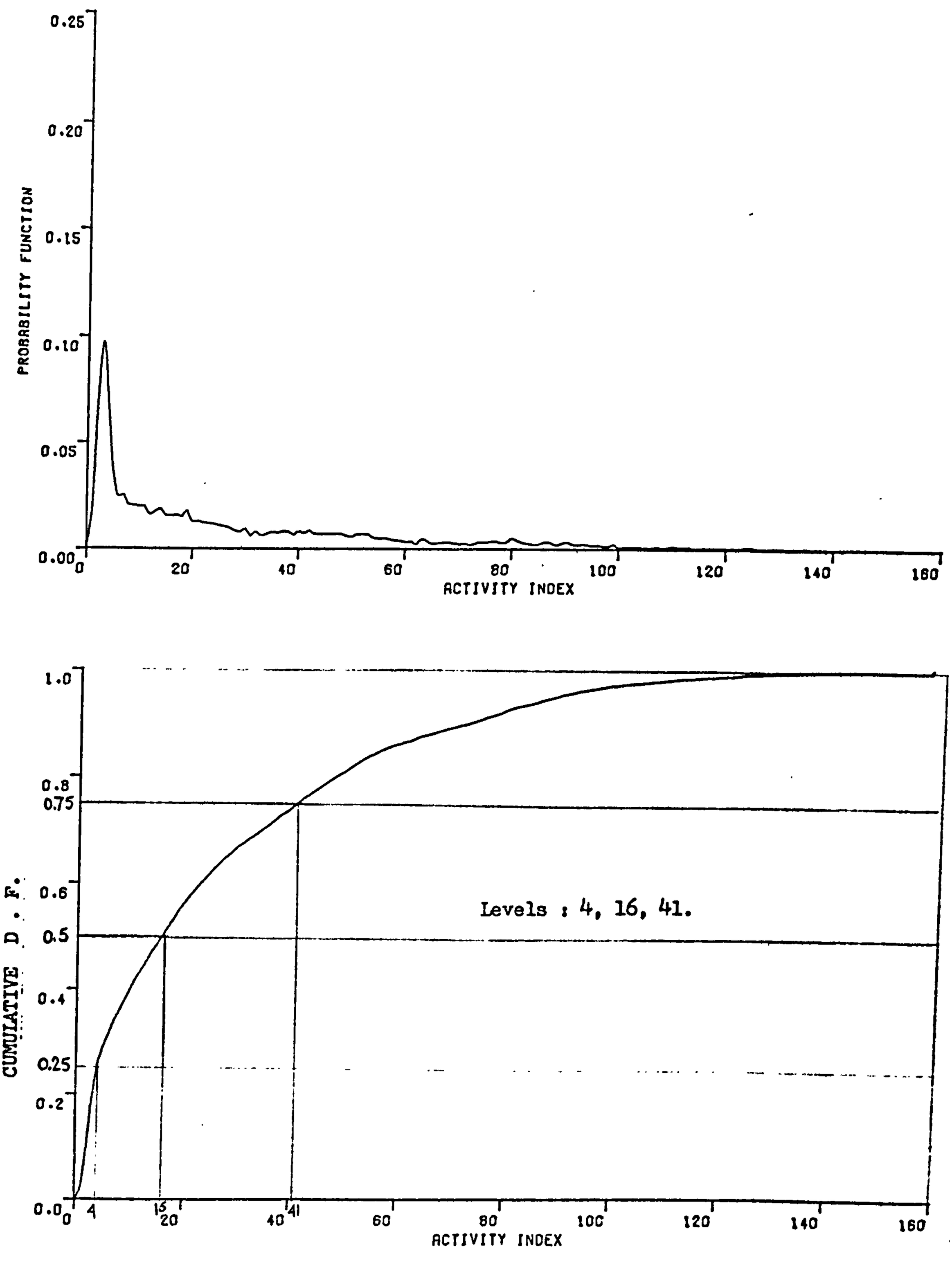


Figure (5.3). Probability functions of activity index for the 'hypothetical' image.

also shown on the graphs. Again, a wide range of differences among different images are observed. Even for the case of 'hypothetical' image, great differences were noted compared with the individual images. These variations may cause total misrepresentation for some subpictures as far as coding is concerned. A subpicture in image (A), for example, may be classified as activity level II, depending on its index value, while the same value of index for a subpicture in image (B) will cause it to be classified as activity level IV. Figure (5.4) summarizes the variations in activity index values among different images and shows the interrelationship among different activity classes in each one.

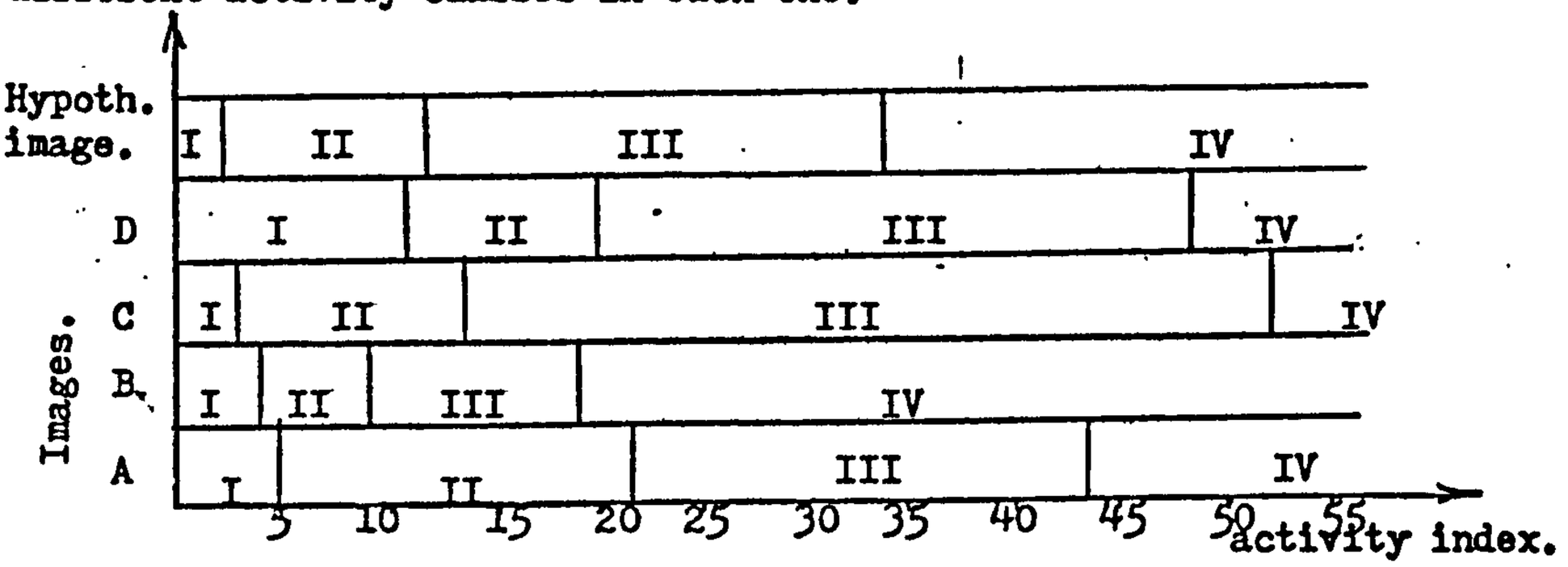


Figure (5.4). Four equally likely activity classes (in Romans), as functions of activity index, for different images.

Another drawback in applying the proposed coding schemes, is the lackof statistical support which justifies the grouping of coefficients in each class. As a result of the over all activity computed in the proposal, he attempted to quantize as many coefficients as possible (in case of class IV), regardless of the different local characteristics of the subpictures.

In anctber work by Chen and Smith[31], the activity index as proposed by Gimlett, was again used in conjunction with Discrete Cosine Transform (DCT). There, again, the subpicture blocks of 16x16 samples were grouped into four classes on the same activity index basis. Four bit

allocation matrices were devised, one for each activity class. Assuming again equal distribution of the different classes, an average bit rate of 1 bit/pixel was demonstrated.

5.3. Directing Index :

In this study, a new " Directing Index " has been devised. This index is supposed to direct each subpicture to its most appropriate coder. It was called Directing Index to emphasize that it is not intended to assess the overall activity of a subpicture, although it does implicitly, but to assess the interrelations among different coefficients within the subpicture transform, and hence, to direct it accordingly.

In setting this index, advantages were taken from some statistical properties in the transform domain, especially the abruptancy and conditional probabilities of coefficients and subsets of coefficients . The sharp decrease in SNR (or any other criterion), as some subpictures deviate from the statistical model (if such a model does really exist), motivated the idea, and hence the decision to care about some minorities among subpictures in the same way as the majority of them. Many different directing indexes have been set and tested. In this Chapter, three are presented as the most applicable and promising ones.

5.3.1. Directing Index D1:

5.3.1.1. Setting up of D1:

In calculating this index, the activity of each subvector is decided by comparing with a preset threshold. The subvector as a whole is considered as low active if its 'activity index' is equal to or less than the threshold, and is considered as highly active, if, on the other hand, its activity index is greater than the preset threshold. As an example, in the case of a 16- pixel vector, the transform domain coefficients are divided into four subvectors : H2, low sequency subvector, medium sequency subvector, and high sequency subvector, as shown in Figure (5.5). Obviously, the coefficient H1 is out of question regarding the procedure of division,

and will not be counted here as a subvector. This is because, as it is always the largest coefficient in all cases, it will escape the act of bit compression, and will be allowed a full 8- bit linear coder to keep the distortion of over all brightness as low as possible.

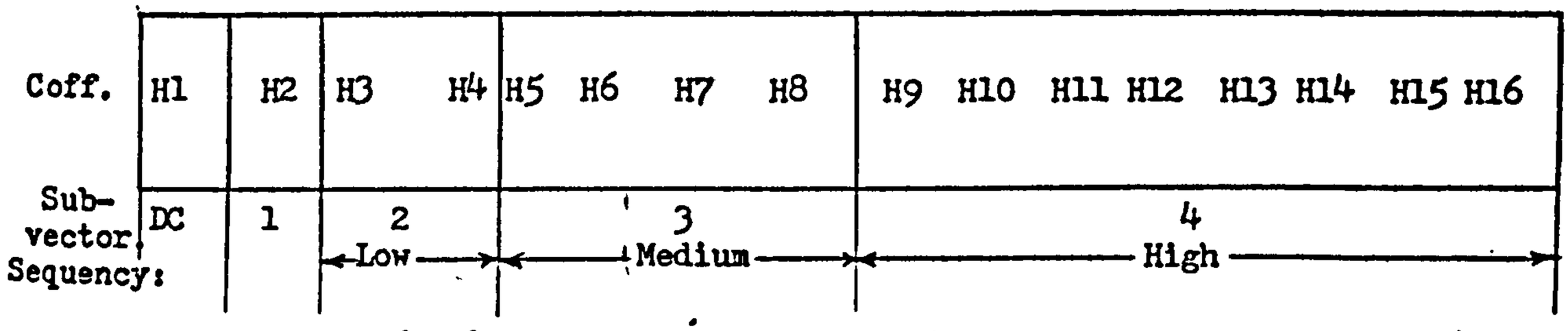


Figure (5.5). Different subvectors of ac coefficients in an N=16vector.

Coefficient H2 will comprise a subvector of its own, as resulted from the binary boundaries. Activity index of this subvector will, of course, be its absolute value |H2| . As for the other three subvectors, activity index for each one will be taken to be the sum of absolute values of all coefficients in the subvector as follows:

$$A = \sum_{i_1}^{i_2} |Hi| \qquad \ldots\ldots (5.3)$$

where i_1 = starting order of coefficients in the subvector,

i_2 = last order in the same subvector.

Next, the three indexes are compared with three thresholds. These thresholds are preset on basis of statistical characteristics of different subvectors. Having decided the activity of each subvector to be Low or High, a combination of all three indexes will decide the over all directing index of the sub-vector. Just as in the case of three binary variables, there will be eight different combinations (2^3).

As an example, let the activity index of a subvector (i) be A_i, and its threshold be $TH_{(i)}$, then the two states of a subvector activity will be:

LOW if $A_{(i)} \leqslant TH_{(i)}$ ..... (5.4)

and HIGH if $A_{(i)} > TH_{(i)}$ .....(5.5)

Assigning a binary weighted measure, an overall Directing Index could be computed using the weights in Table (5.1).

Table (5.1). Weights of activities for different subvectors.

Subvector 2 Low seq. coeffs. H3 - H4		Subvector 3 Medium seq. coeffs. H5 - H8		Subvector 4 High seq. coeffs. H9 - H16	
LOW	HIGH	LOW	HIGH	LOW	HIGH
0	4	0	2	1	2

The directing index will then be the total sum of these corresponding weights. For example, assume that for a given vector:

$A_2 < TH_{(2)}$ (LOW),

$A_3 > TH_{(3)}$ (HIGH),

and $A_4 > TH_{(4)}$ (HIGH),

then the corresponding weights will be 0, 2, and 2 respectively. Thus, the directing index for this particular vector will be

INDEX D1 =0 + 2 + 2 = 4.

A coder associated with index D1= 4 should therefore be suitable for a low active (H3 + H4), highly active medium sequency- and a highly active high sequency subvector areas.

Table (5.2) shows the whole combinations for different values of Directing Index D1.

The thresholds for different subvectors were selected, as mentioned before, on statistical bases and were slightly modified after that to get the best possible results. A separate coder was set for each particular index value. These coders were all based on $Max^{(60)}$ type quantizers.

Some changes in values have been tried to get the maximum possible fidelity.

Table (5.2). Directing Index D1 for different combinations of subvectors activity levels.

Subvector activity			Weights.			Directing Index D1
2nd	3rd	4th	2nd	3rd	4th	
L	L	L	0	0	1	1
L	L	H	0	0	2	2
L	H	L	0	2	1	3
L	H	H	0	2	2	4
H	L	L	4	0	1	5
H	L	H	4	0	2	6
H	H	L	4	2	1	7
H	H	H	4	2	2	8

5.3.1.2. Bit Allocation and Complete Coders for D1:

Table (5.3) summarizes the number of bits assigned to each ac coefficient for all different values of Directing Index D1, subject to the arbitrary bit rate of 2 bits/sample (pixel). From table, number of bits allocated to some coefficients are decreased when their standard deviations are small, while some others could be discarded altogether. For increased values of standard deviations, the number of bits are also increased. The set of thresholds used was 9, 12, and 12 respectively.

Table (5.3). Number of bits allocated for each ac coefficient for a bit rate of 2bits/sample.

D1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16
1.	3	3	3	3	2	2	2	2	2	2	0	0	0	0	0
2	3	3	2	0	0	0	0	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	2	0	0	0	0	0	0	0
4	4	2	2	2	2	2	2	2	2	2	2	0	0	0	0
5	4	4	4	2	2	2	2	2	2	0	0	0	0	0	0
6	4	4	4	0	0	0	0	2	2	2	2	2	0	2	0
7	4	4	4	3	3	3	3	0	0	0	0	0	0	0	0
8	4	2	2	2	2	2	2	2	2	2	2	0	0	0	0

Coders to be used with Directing Index D1 are shown, in full, in Tables (5.4 - 5-11).

Note: in coders of Tables (5.4 - 5.11) and all coders thereafter, the values shown are such that a representative R_i is assigned for the range of values equal to or larger than a quantizing level Q_i and less than the next level Q_{i+1}, as indicated below :

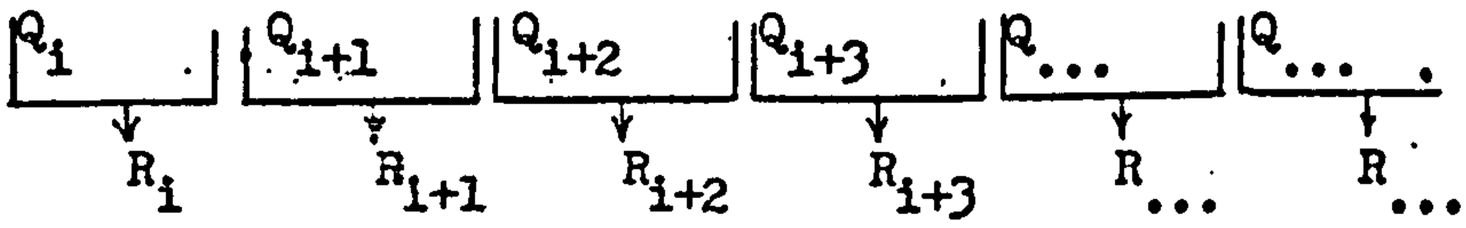


Table (5.4). Coders used with Directing Index D1= 1.

H2	Quant. levels	+ 0 3 5 9 256
	Representatives	− 1 3 6 11
H3	Quant. levels	+ 0 2 4 6 256
	Representatives	− 0 2 4 7
H4	Quant. levels	+ 0 2 4 6 256
	Representatives	− 0 2 4 7
H5	Quant. levels	+ 0 2 4 6 256
	Representatives	− 0 2 4 7
H6	Quant. levels	+ 0 3 256
	Representatives	− 1 5
H7	Quant. levels	+ 0 3 256
	Representatives	− 1 5
H8	Quant. levels	+ 0 3 256
	Representatives	− 1 5
H9	Quant. levels	+ 0 3 256
	Representatives	− 1 4
H10	Quant. levels	+ 0 3 256
	Representatives	− 1 4
H11	Quant. levels	+ 0 3 256
	Representatives	− 1 4
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.5). Coders used with Directing Index D1= 2.

H2	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H3	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H4	Quant. levels	± 0 2 256
	Representatives	± 0 3
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 3 256
	Representatives	± 1 5
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	± 0 3 256
	Representatives	± 1 5
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	± 0 3 256
	Representatives	± 1 5
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	± 0 3 256
	Representatives	± 1 5
H16	Quant. levels	± 0 3 256
	Representatives	± 1 5

Table (5.6). Coders for use with Directing Index D1= 3.

H2	Quant. levels	+ 0 3 5 9 256
	Representatives	− 1 3 6 10
H3	Quant. levels	+ 0 3 5 9 256
	Representatives	− 1 3 6 10
H4	Quant. levels	+ 0 3 5 9 256
	Representatives	− 1 3 6 10
H5	Quant. levels	+ 0 8 16 26 256
	Representatives	− 3 11 20 32
H6	Quant. levels	+ 0 8 16 26 256
	Representatives	− 3 11 20 32
H7	Quant. levels	+ 0 8 16 26 256
	Representatives	− 3 11 20 32
H8	Quant. levels	+ 0 7 15 24 256
	Representatives	− 3 11 19 30
H9	Quant. levels	+ 0 2 256
	Representatives	− 0 3
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.7). Coders for use with Directing Index D1= 4.

H2	Quant. levels	± 0 4 8 12 16 22 28 256
	Representatives	± 1 5 9 13 19 24 31
H3	Quant. levels	± 0 3 256
	Representatives	± 1 5
H4	Quant. levels	± 0 3 256
	Representatives	± 1 5
H5	Quant. levels	± 0 6 256
	Representatives	± 2 8
H6	Quant. levels	± 0 6 256
	Representatives	± 2 8
H7	Quant. levels	± 0 6 256
	Representatives	± 2 8
H8	Quant. levels	± 0 6 256
	Representatives	± 2 8
H9	Quant. levels	± 0 6 256
	Representatives	± 2 8
H10	Quant. levels	± 0 6 256
	Representatives	± 2 8
H11	Quant. levels	± 0 6 256
	Representatives	± 2 8
H12	Quant. levels	± 0 6 256
	Representatives	± 2 8
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.8). Coders for use with Directing Index D1=5.

H2	Quant. levels	± 0 4 8 12 16 22 28 256
	Representatives	± 1 5 9 13 19 25 31
H3	Quant. levels	± 0 4 8 12 16 22 28 256
	Representatives	± 1 5 9 13 19 24 31
H4	Quant. levels	± 0 4 8 12 16 22 28 256
	Representatives	± 1 5 9 13 19 24 31
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 256
	Representatives	± 1 5
H9	Quant. levels	± 0 3 256
	Representatives	± 1 5
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.9). Coders for use with Directing Index D1= 6.

H2	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H3	Quant. levels	± 0 3 6 10 13 17 22 29 256
	Representatives	± 1 4 8 11 15 19 25 33
H4	Quant. levels	± 0 3 6 10 13 17 22 29 256
	Representatives	± 1 4 8 11 15 19 25 33
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 5 256
	Representatives	± 2 8
H10	Quant. levels	± 0 5 256
	Representatives	± 2 8
H11	Quant. levels	± 0 5 256
	Representatives	± 2 8
H12	Quant. levels	± 0 5 256
	Representatives	± 2 8
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 5 256
	Representatives	± 2 8
H16	Quant. levels	-
	Representatives	-

Table (5.10). Coders for use with Directing Index D1= 7.

H2	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H3	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H4	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H5	Quant. levels	± 0 6 13 21 256
	Representatives	± 2 9 16 24
H6	Quant. levels	± 0 6 13 21 256
	Representatives	± 2 9 16 24
H7	Quant. levels	± 0 6 13 21 256
	Representatives	± 2 9 16 24
H8	Quant. levels	± 0 6 13 21 256
	Representatives	± 2 9 16 24
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.11). Coders for use with Directing Index DI= 8.

H2	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H3	Quant. levels	± 0 3 256
	Representatives	± 1 5
H4	Quant. levels	± 0 3 256
	Representatives	± 1 5
H5	Quant. levels	± 0 6 256
	Representatives	± 2 9
H6	Quant. levels	± 0 6 256
	Representatives	± 2 9
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	± 0 6 256
	Representatives	± 2 9
H9	Quant. levels	± 0 6 256
	Representatives	± 2 9
H10	Quant. levels	± 0 6 256
	Representatives	± 2 9
H11	Quant. levels	± 0 6 256
	Representatives	± 2 9
H12	Quant. levels	± 0 6 256
	Representatives	± 2 9
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

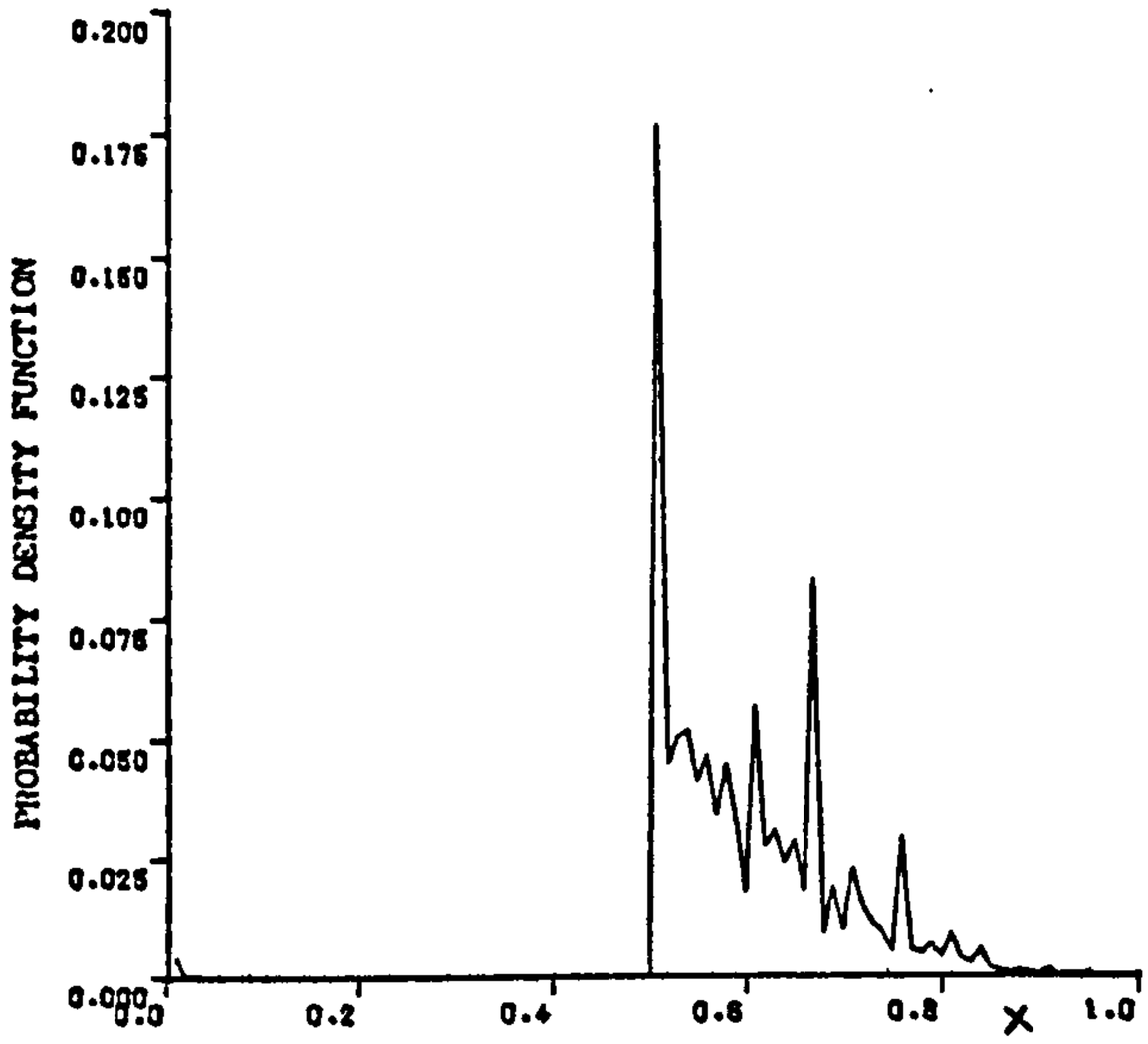
5.3.2. Directing Index D2 :

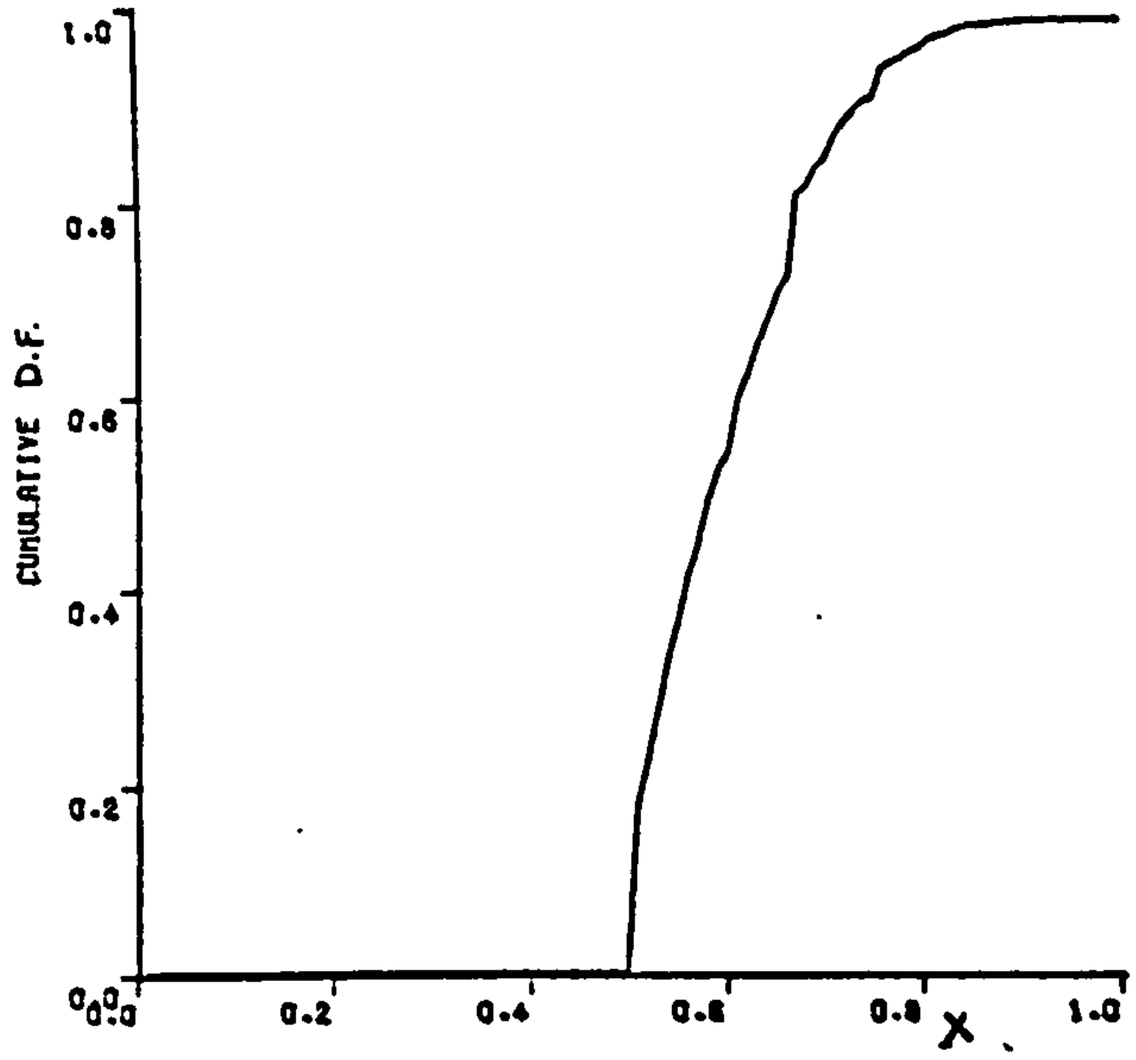
In this index, the interrelation between medium- and high-sequency subvectors is used to control the selection of coefficients to be coded from each subvector. It has been found, statistically, that a considerable proportion of vectors have their high sequency energy content associated with either odd or even ordered coefficients within the subvector. Hence, in this indexing procedure, both the medium sequency subvector (H5 - H8), and the high sequency subvector (H9 - H16), are further divided into two subsets each. The first subset of each subvector contains all the odd numbered coefficients , while the second contains all the even numbered coefficients.

5.3.2.1. Energy Spectrum and D2 Set-up :

Taking only one subset (either odd or even numbered coefficients), and calculating the ac energy associated with it, and then relating this to the total ac energy content in the corresponding subvector has shown some interesting results. As an example, Figure (5.6) shows the probability functions of either subset (in the high sequency subvector) having a proportion of the total high sequency subvector energy content. The first feature to be noted, is the general similarity in profile among different images. The second important feature is the sharp clustering at particular values of energy content proportion. As shown in the graphs, these needle like spikes appear more or less at the ranges of 0.5 ,0.61, (0.68 - 0.71), and 0.77 of total energy content.

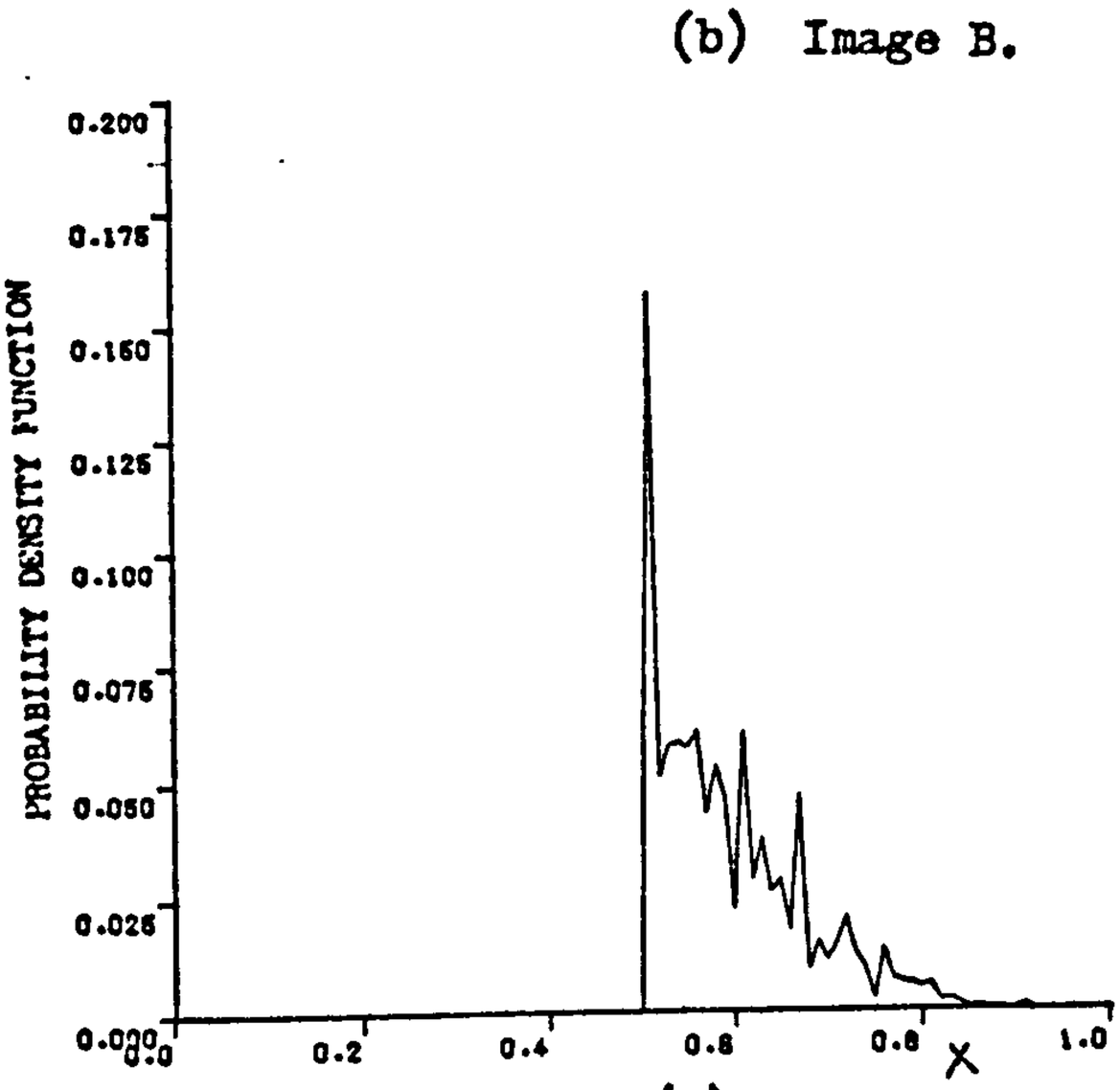
For those vectors which have a great proportion of the medium- or high sequency energy content clustered in one of the two subsets (odd or even), it will be more realistic to code that particular subset instead of neglecting the entire subvector (as in the case of high sequency), or coding the whole subvector (as in the case of medium sequency).

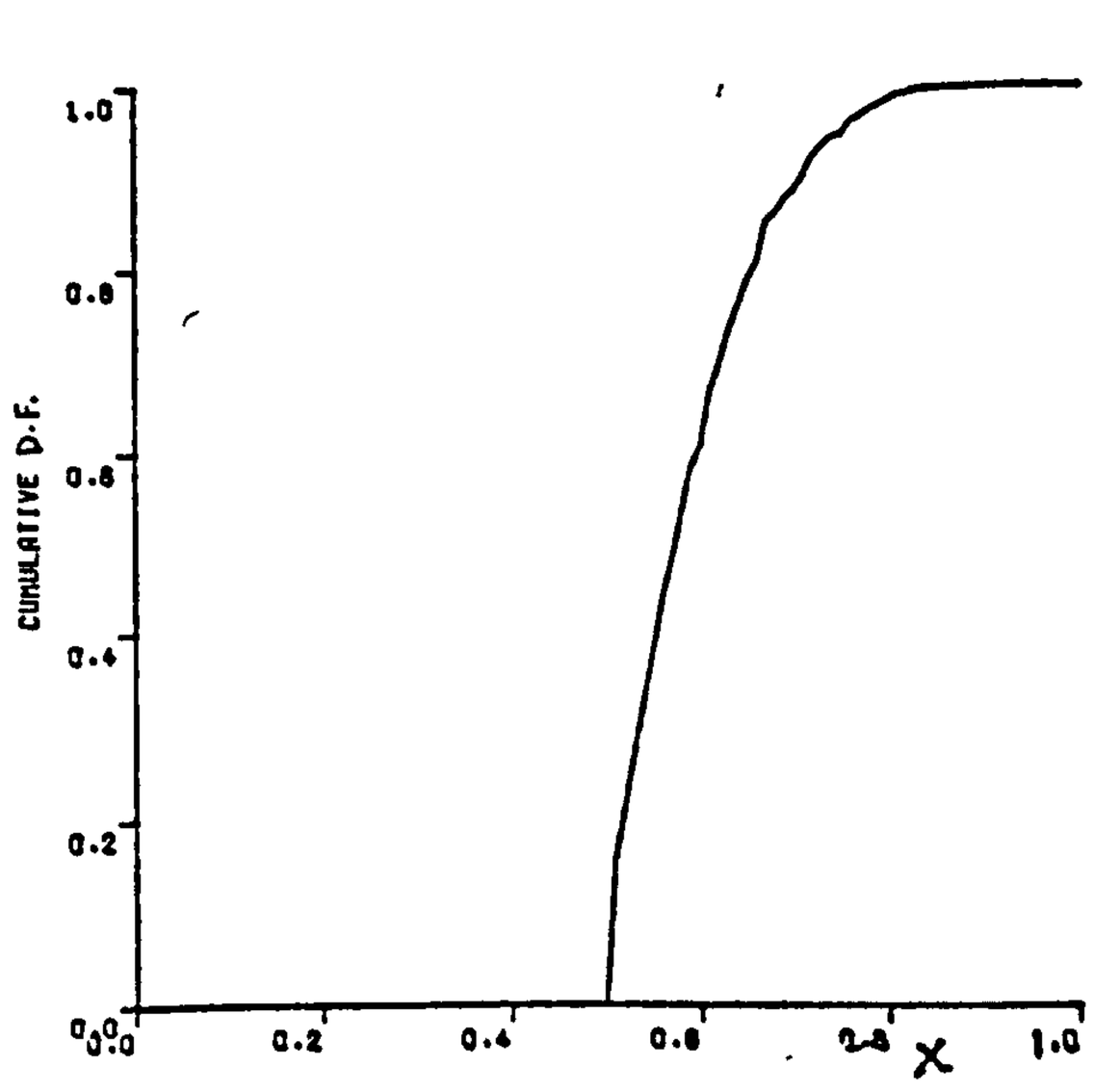




(a) Image A.

(b) Image B.





(c) Image D.

x= proportion of energy contenet in the most active subset.

Figure (5.6). Probability of energy clustering in either odd or even subsets.

Let the high sequency subvector be divided as:

$$[High]_{odd} = [H9, H11, H13, H15] \quad \text{..... odd subset} \quad (5.6)$$

and $$[High]_{even} = [H10, H12, H14, H16] \quad \text{..... even subset} \quad (5.7)$$

As a measure of each subset activity, again the sum (SH) of the absolute values of coefficients in the subset is taken :

$$SH\ ODD = \sum |Hi| \qquad i=9, 11, 13, 15 \quad \text{.....} \quad (5.8)$$

$$SH\ EVEN = \sum |Hi| \qquad i=10, 12, 14, 16 \quad \text{.....} \quad (5.9)$$

An overall high sequency subvector activity measure is then defined as the maximum value between these two measures:

$$h = Max\ (SH\ ODD\ ,\ SH\ EVEN) \quad \text{.....} \quad (5.10)$$

Following a similar procedure for the medium sequency subvector, the odd and even numbered subsets' activity measures will be:

$$SM\ ODD = \sum |Hi| \qquad i=5,7 \quad \text{.....} \quad (5.11)$$

and $$SM\ EVEN = \sum |Hi| \qquad i=6,8 \quad \text{.....} \quad (5.12)$$

and an overall medium sequency subvector activity measure will, by analogy, be taken to represent the whole subvector activity as:

$$m = Max\ (SM\ ODD\ ,\ SM\ EVEN) \quad \text{.....} \quad (5.13)$$

By such procedures, the most active half of each subvector has been selected to be coded. Obviously, the number of possible different combinations in such a scheme will be 4. But to allow the index to consider also the two extreme cases where either the subvectors has no (or very little) activity, another comparative measure is set. This is taken to be the ratio between the computed activities of the two subvectors:

$$BALANCE\ B = h/m \quad \text{. . . . . .} \quad (5.14)$$

When the balance B is less than a preset minimum threshold value TMIN, the high sequency subvector is considered as low active, and then no coefficient of this subvector is coded at all. All the bits saved in this case will be re-allocated to code all the coefficients in the medium sequency subvector. More bits can also be re-allocated to coefficients

in low-sequency subvector.

On the other hand, in the second extreme case where no (or very few) details are present in the medium sequency subvector, the balance B will be greater than a preset maximum threshold value TMAX. Advantage will be taken of that, and a higher number of coefficients in the high sequency subvector will be coded on the expense of no coding at all in the medium sequency subvector. In this present example, 6 out of 8 coefficients have been coded in such a case. Naturally, which particular two coefficients are to be left uncoded, is a matter of experiment and statistical studies. It has been found (by simulation results in this example) that the most appropriate solution is todiscard either H14 & H16, or H13 & H15 .

Another important case which arises nearly 40 % of total cases, is when no details are present either in the medium sequency or in the high sequency subvectors. This is a DC, or near DC, case, and the vector is directed to any coder which codes the low sequency coefficients more accurately.

Proportions of vectors directed to each coder, as well as the overall fidelity, will of course be functions of the thresholds TMIN and TMAX with which the balance B is compared. Extensive simulation experiments were performed, during which TMIN and TMAX were varied. It has been found that reasonable values for these thresholds are as follows:

TMIN = 0.4 - 0.6

and TMAX = 1.6 - 2.0

Computation of Index D2 :

Computation of index D2 is done in two steps. In the first, a basic index is computed, while a modification index is computed in the second.

A. Basic index.

The balance measure B between medium and high sequency subvectors is calcultaed as mentioned earlier. It is then compared with TMIN and TMAX. The basic index values, depending on the comparison result , are as shown in Table (5.12).

Table (5.12). Basic index depending on B.

Comparison result.	Basic index.
$B \leqslant TMIN$	1
$B \geqslant TMAX$	2
$TMIN < B < TMAX$	3

B. Modification index.

If the basic index value is 3, then another index, called modification index is added to it. This index is shown in Table (5.13), depending on which subset of each subvector is the most effective.

Table (5.13). Modification index values .

The most active subset in each subvector.		Modification index value.
MEDIUM SEQ.	HIGH SEQ.	
ODD	ODD	0
ODD	EVEN	1
EVEN	ODD	2
EVEN	EVEN	3

Finally, the Directing Index D2 will be:

D2 = Basic index + Modification index
(if the basic index is 3).

5.3.2.2. Bit Allocation and Complete Coders for D2 :

Table (5.14) shows different combinations discussed above, and the corresponding values of directing index D2. It also shows the number of bits allocated for each coefficient .

Table (5.14). Number of bits allocated for each coefficient in conjunction with D2.

D2	Features of vector. Order		Number of bits for coefficients. 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
1	B ≤ TMIN		4 4 4 3 3 3 3 0 0 0 0 0 0 0 0
2	B ≥ TMAX		4 4 4 0 0 0 0 2 2 2 2 2 0 2 0
	TMIN < B < TMAX		
	Med. seq.	High seq.	
3	ODD	ODD	4 3 3 3 0 3 0 2 0 2 0 2 0 2 0
4	ODD	EVEN	4 3 3 3 0 3 0 0 2 0 2 0 2 0 2
5	EVEN	ODD	4 3 3 0 3 0 3 2 0 2 0 2 0 2 0
6	EVEN	EVEN	4 3 3 0 3 0 3 0 2 0 2 0 2 0 2

Coders to be used with Directing Index D2 are shown, in full, in Tables (5.15 - 5.20).

Table (5.15). Coders for use with Directing Index D2= 1.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H4	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H5	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H6	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H7	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H8	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.16). Coders for use with Directing Index D2= 2.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H4	Quant. levels	± 0 1 2 3 4 5 6 8 10 256
	Representatives	± 0 1 2 3 4 5 6 8 11
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 3 256
	Representatives	± 1 4
H10	Quant. levels	± 0 3 256
	Representatives	± 1 4
H11	Quant. levels	± 0 3 256
	Representatives	± 1 4
H12	Quant. levels	± 0 3 256
	Representatives	± 1 4
H13	Quant. levels	± 0 3 256
	Representatives	± 1 4
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 3 256
	Representatives	± 1 4
H16	Quant. levels	-
	Representatives	-

Table (5.17). Coders for use with Directing Index D2= 3.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 3 256
	Representatives	± 1 4
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	± 0 3 256
	Representatives	± 1 4
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	± 0 3 256
	Representatives	± 1 4
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 3 256
	Representatives	± 1 4
H16	Quant. levels	-
	Representatives	-

Table (5.18). Coders for use with Directing Index D2= 4.

H2	Quant. levels	± 0 1 3 5 7 9 11 14 256
	Representatives	± 0 1 3 5 7 9 12 15
H3	Quant. levels	± 0 3 5 9 256
	Representatives	± 1 3 6 11
H4	Quant. levels	± 0 3 5 9 256
	Representatives	± 1 3 6 11
H5	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H6	Quant. levels	-
	Representatives	- !
H7	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H1C	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	± 0 3 256
	Representatives	± 1 5

Table (5.19). Coders for use with Directing Index D2= 5.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 6
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	± 0 3 256
	Representatives	± 1 4
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	± 0 3 256
	Representatives	± 1 4
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 3 256
	Representatives	± 1 4
H16	Quant. levels	-
	Representatives	-

Table (5.20). Coders for use with Directing Index D2= 6.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H9	Quant. levels	-
	Representatives	-
H1C	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	± 0 3 256
	Representatives	± 1 5

5.3.3. Directing Index D3 :

In this index, all the coefficients of the medium sequency subvector are coded on the expense of slightly decreased accuracy in coding of the first coefficient H1,(the dc term). For the high sequency subvector, although still half the coefficients are coded, this index allows more freedom in selecting the representing subset of coefficients.

Illustrating by the same example as before, the procedures for this index are as follows :

Coefficients of the high sequency subvector are divided into four subsets. In addition to the two subsets mentioned in conjunction with the procedure of the previous directing index D2, there are two other new subsets. One comprises the coefficients of lower-half sequency and the other contains the coefficients in the upper-half sequency of the subvector. These two new subsets are defined as :

$$[High]_{lower} = [H9, H10, H11, H12] \quad \ldots\ldots\ldots\ldots\ldots (5.15)$$

and $$[High]_{upper} = [H13, H14, H15, H16] \quad \ldots\ldots\ldots\ldots\ldots (5.16)$$

As a measure of each subset activity, the two corresponding new measures are introduced here as :

$$SHLOW = \sum_{i=9}^{i=12} |Hi| \quad \ldots\ldots\ldots\ldots\ldots (5.17)$$

$$SHUP = \sum_{i=13}^{i=16} |Hi| \quad \ldots\ldots\ldots\ldots\ldots (5.18)$$

By analogy with previous index D2, and adding these two new measures, an overall high sequency subvector activity measure is set as

$$h = Max\ (\ SHLOW, SHUP, SHODD, SHEVEN\ ) \quad \ldots\ldots\ldots (5.19)$$

For the medium sequency subvector, as all the coefficients will be coded in each case, the overall activity measure for this subvector should be modified to be the total sum of absolute values of all coefficients in the subvector , as follows

$$m = \sum_{i=5}^{i=8} |Hi| \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.20)$$

Computing the balance B, and directing procedures are almost the same as in the case of index D2.

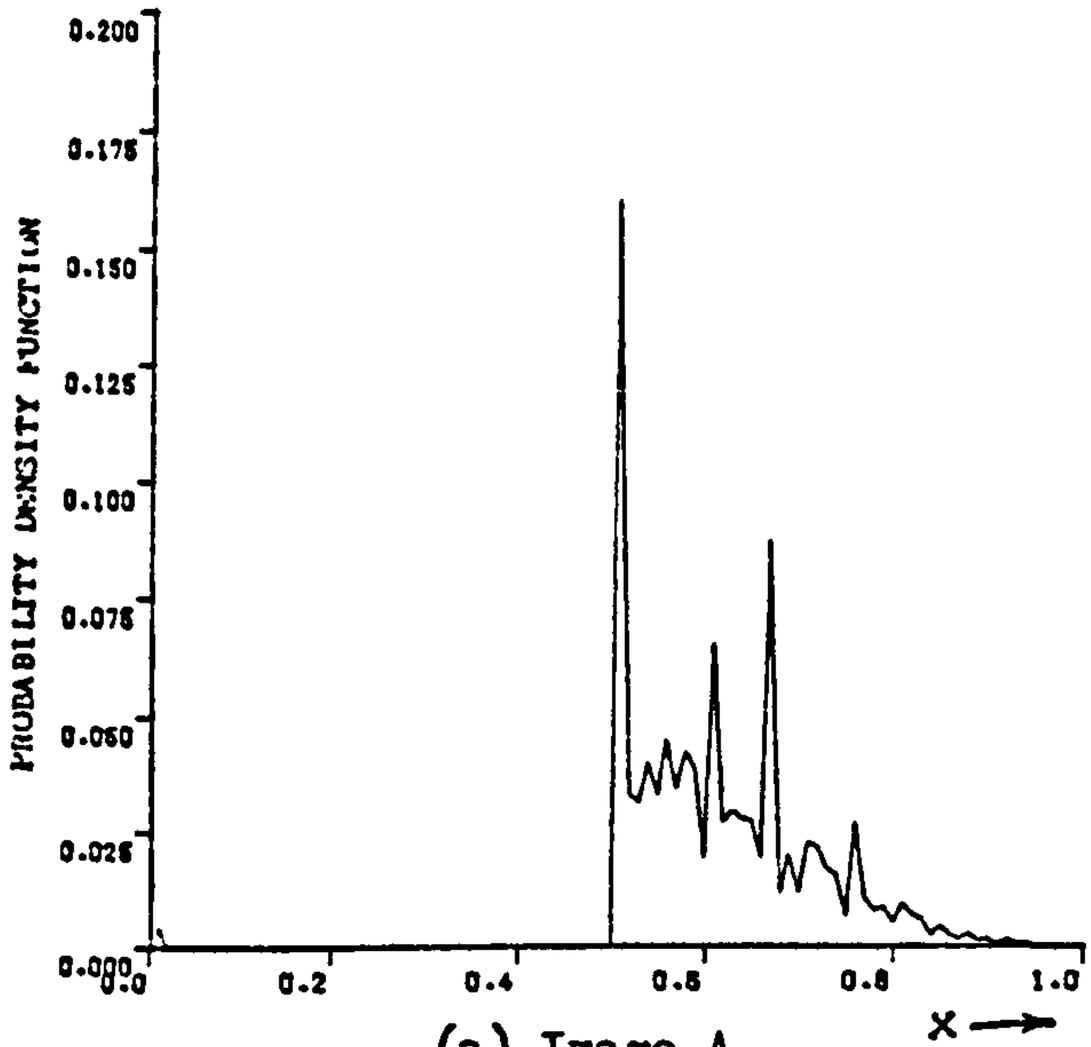
However, to allow continuous coding of all the medium sequency coefficients,some means of saving bits are then necessary. Firstly, the medium sequency coefficients are coded more coarsely than before. Secondly, the first coefficient, Hl, is coded with a 6-bit coder instead of the usual 8-bit one used so far. Again, in the case where no details are present in the high sequency subvector (i.e. when B<TMIN), fine coding is allowed for medium and low sequency coefficients, as well as a full 8-bit coding is allowed for Hl. In the other case, where no medium sequency details are present, the bits normally allocated for them will be re-allocated for more high sequency coefficients. In the example shown here, it is again 6 coefficients out of 8.

Experimental simulations have shown that the greater choice in deciding the most active subset of the high sequency subvector, by increasing the number of these subsets, more than compensated the reduced accuracy in coding Hl and some other low and medium sequency coefficients.

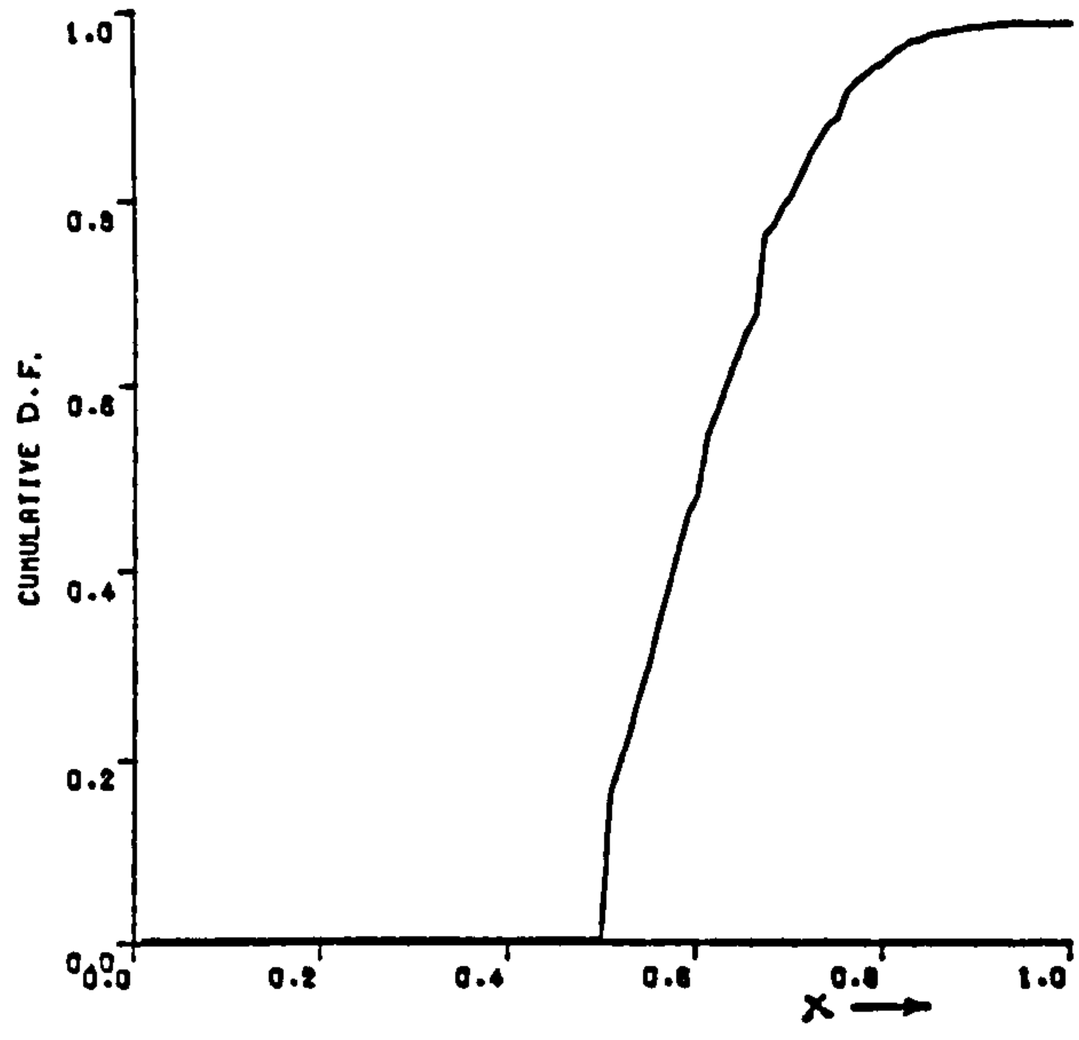
5.3.3.1. Energy Spectrum of High Sequency Subvector and Set up of D3:

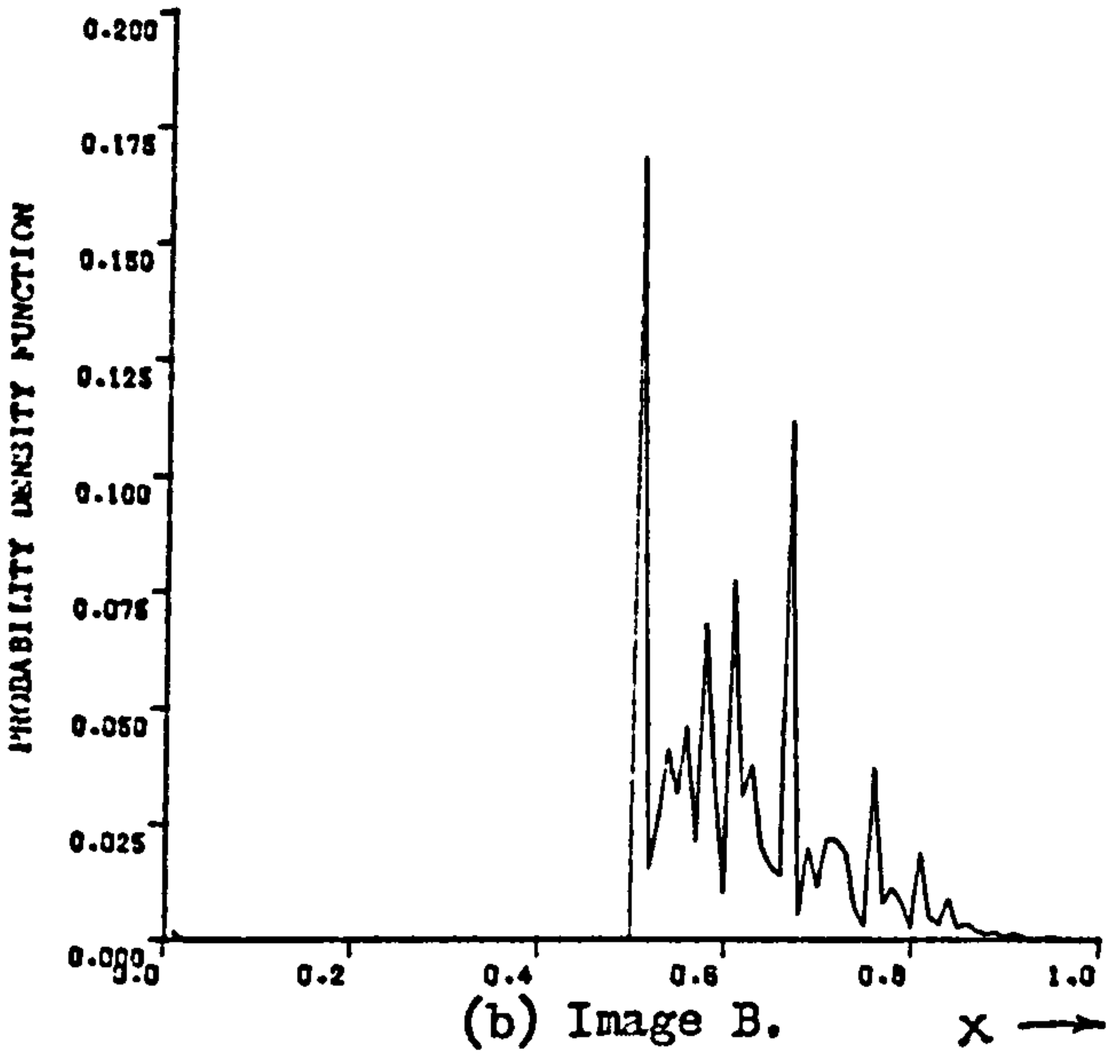
Figures (5.7) show the probability of either subset (lower or upper seq.halves), having a proportion of the total high sequency sub-vector energy content. As in the case of Figures (5.6), the horizontal axis shows the ratio of energy content in either one of the subsets.

Figure (5.8) shows the probability of either one of the four subsets, (lower,upper, odd, or even) having a proportion of the total high sequency subvector energy. Compared with the two sets of graphs (5.6 and 5.7), the most important different feature is the noticeable drop in the prob-ability at the critical value of energy ratio of 0.5 by about 50 %

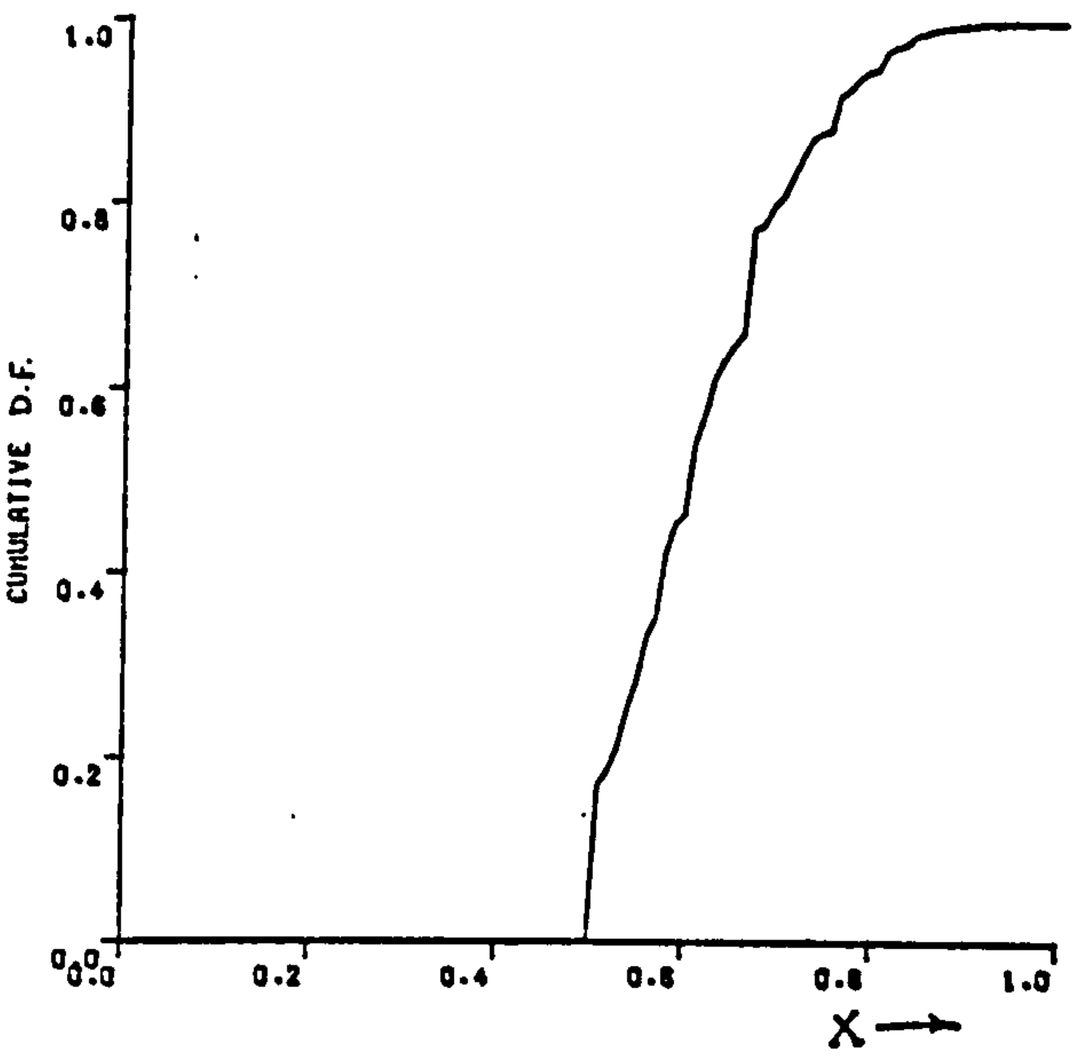


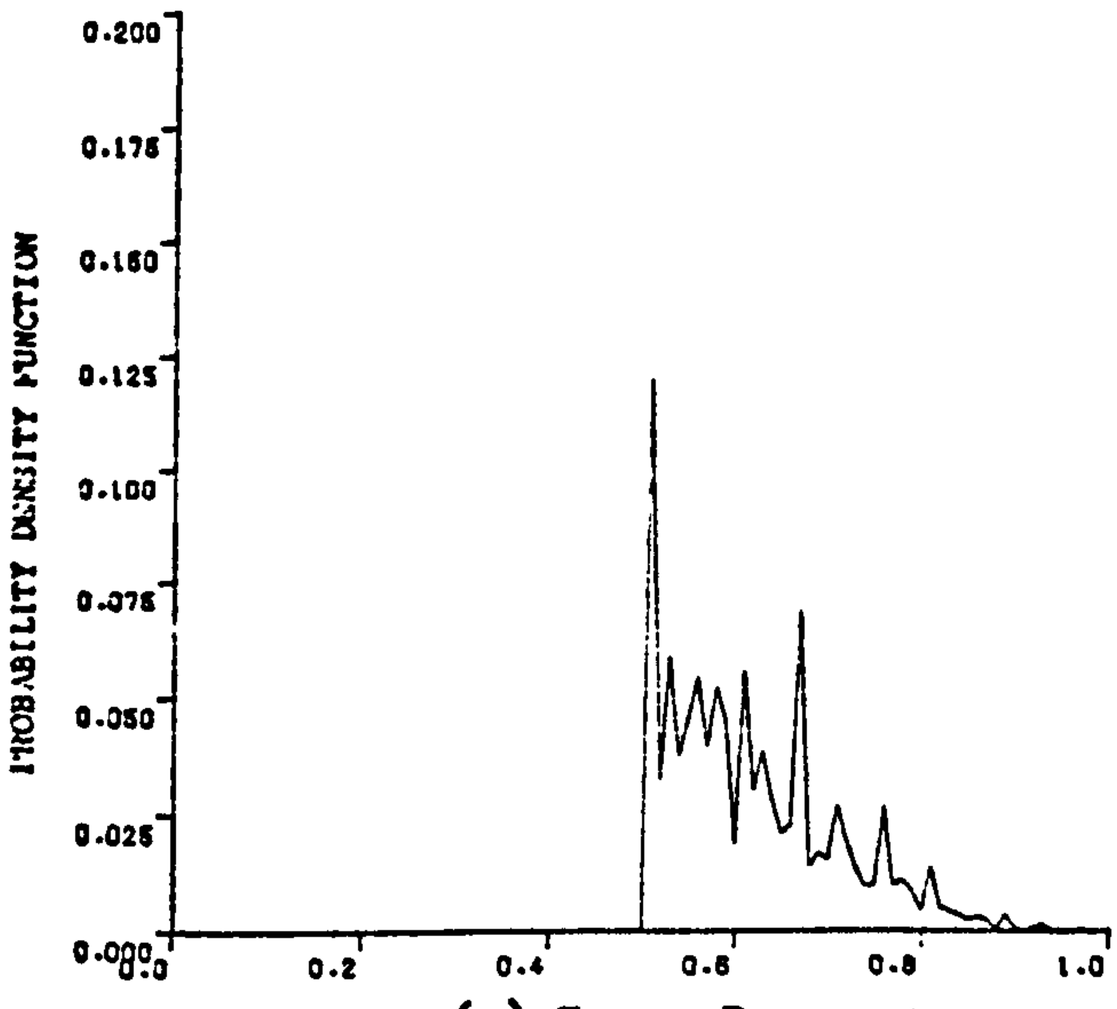
(a) Image A



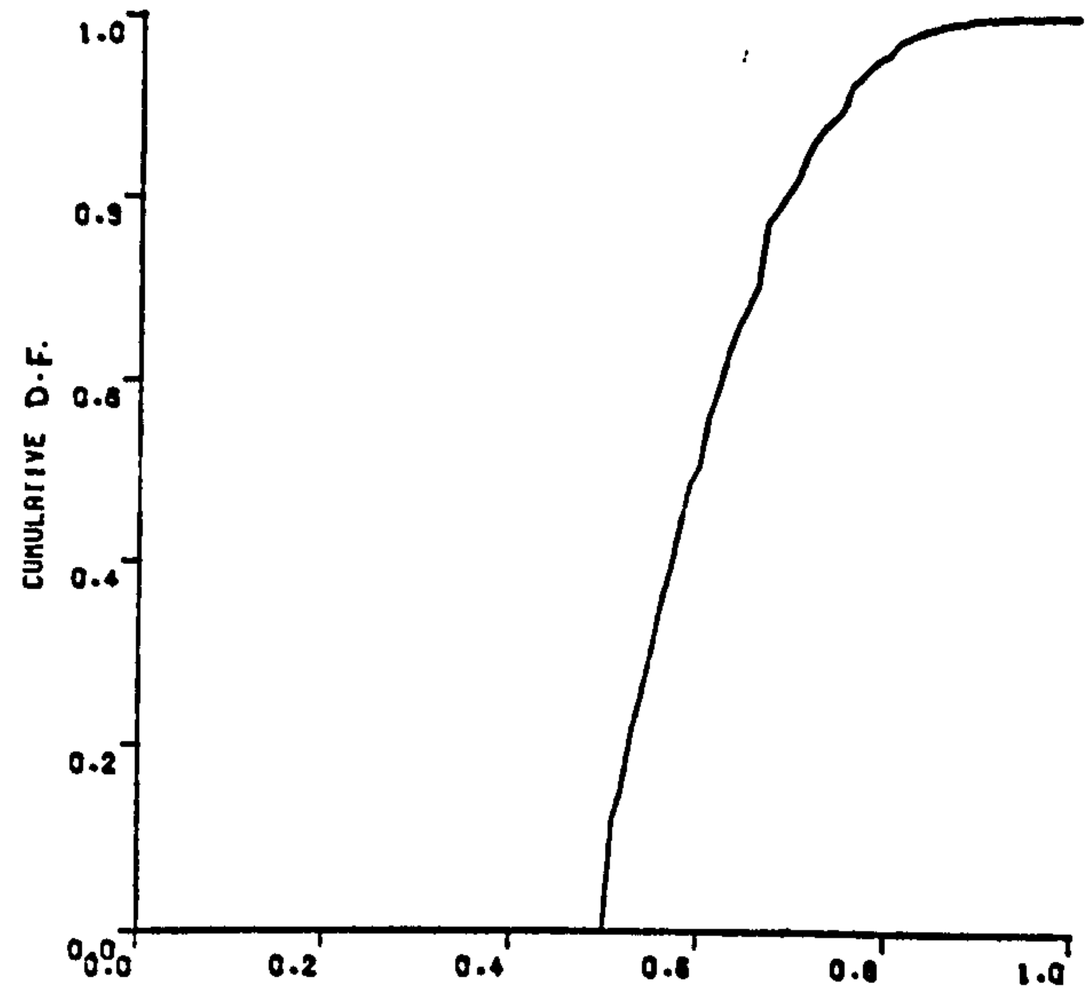


(b) Image B.



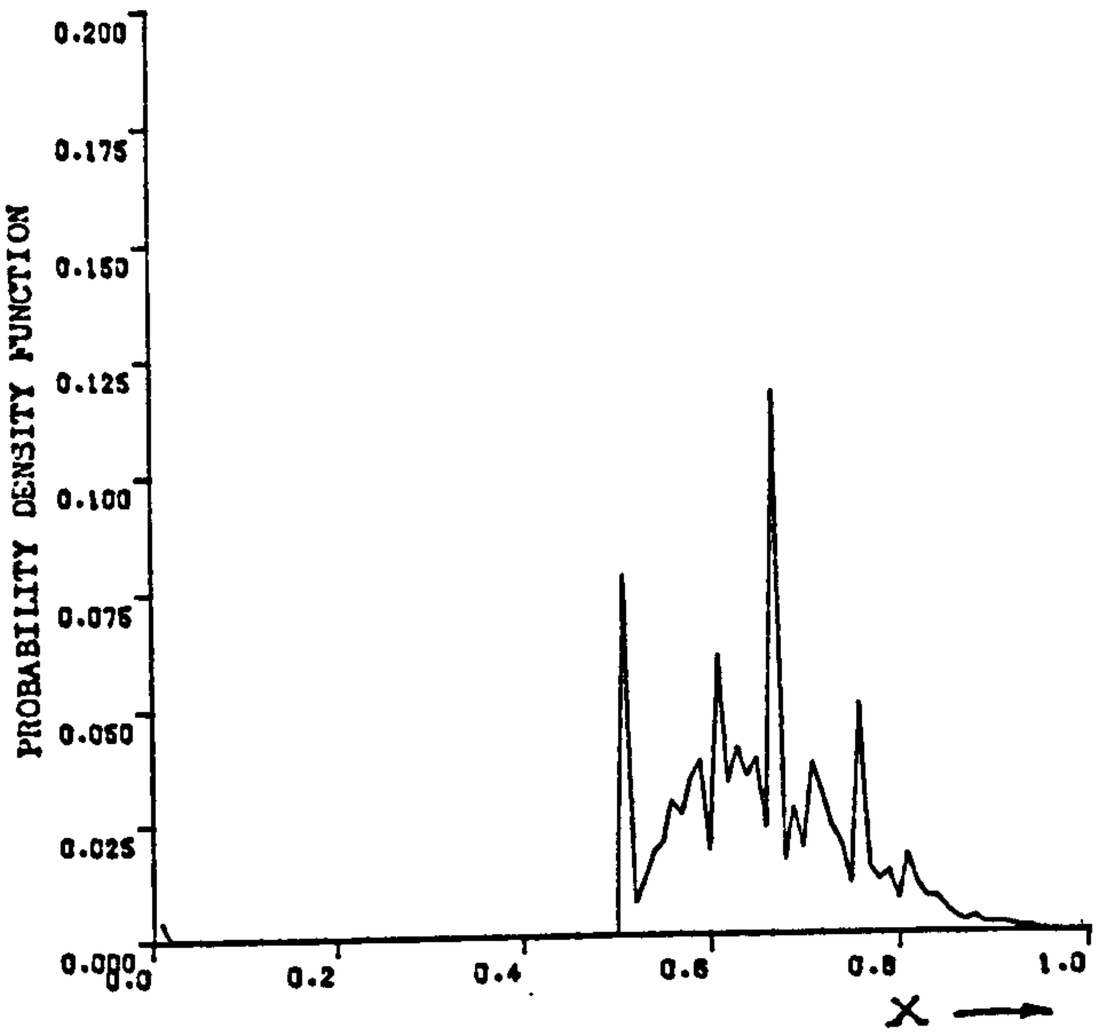


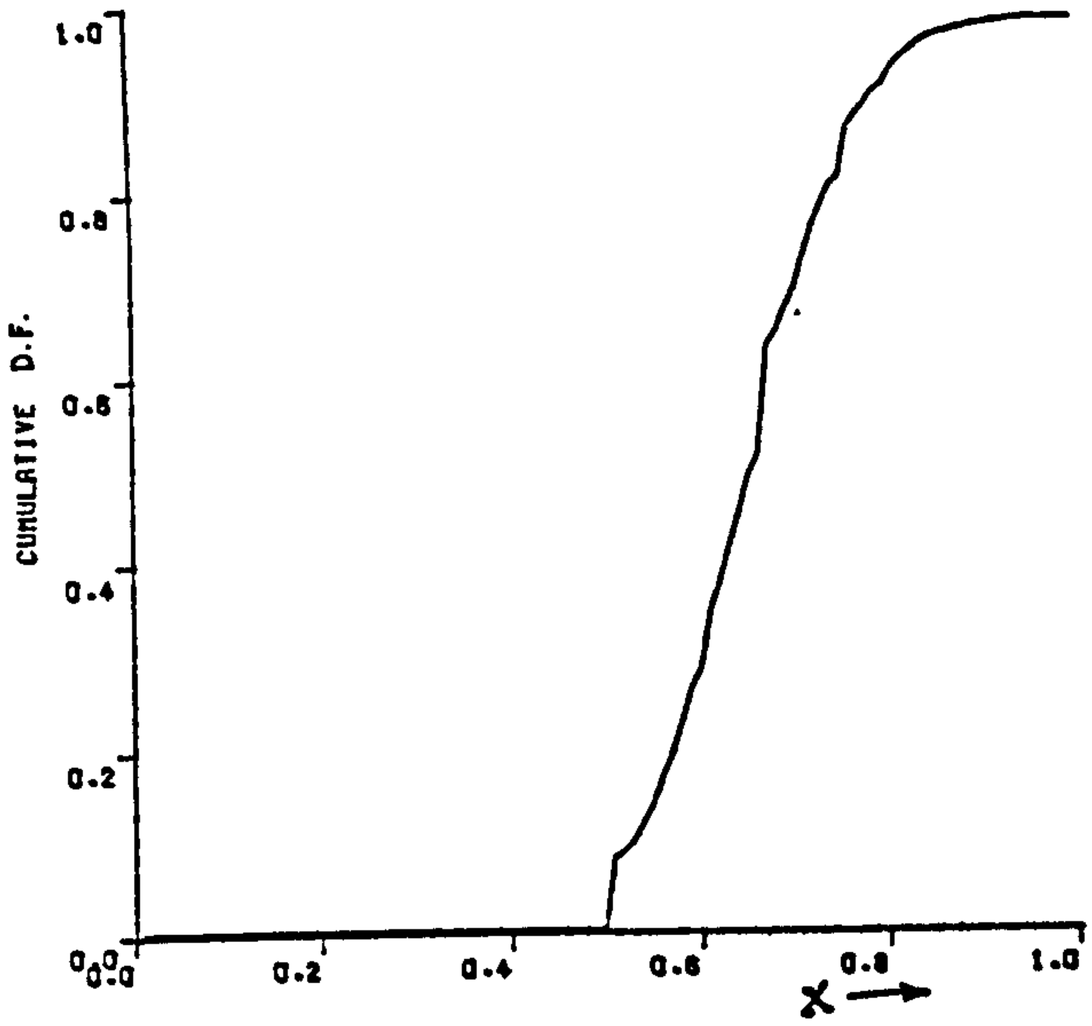
(c) Image D.



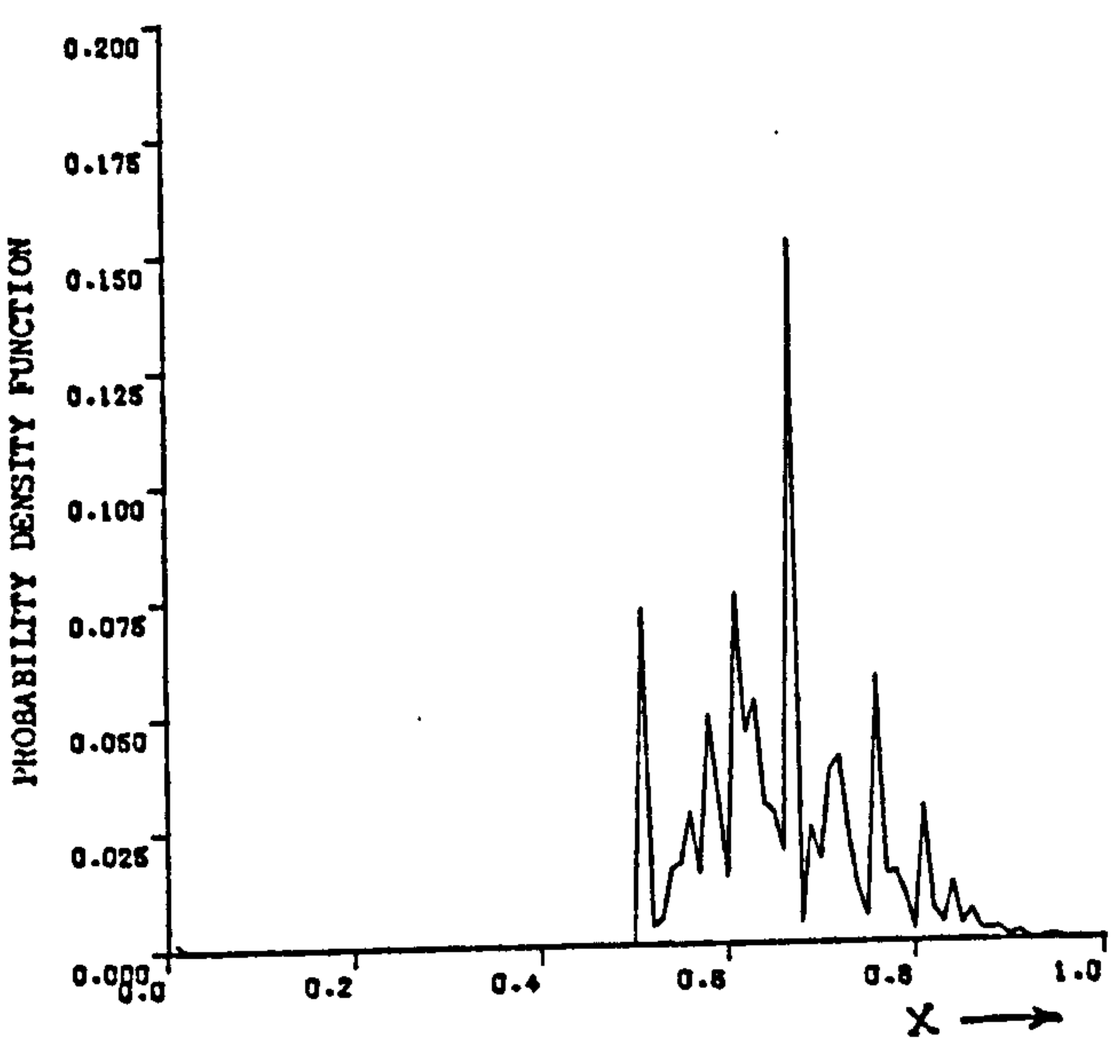
x= proportion of energy content in the most active subset.

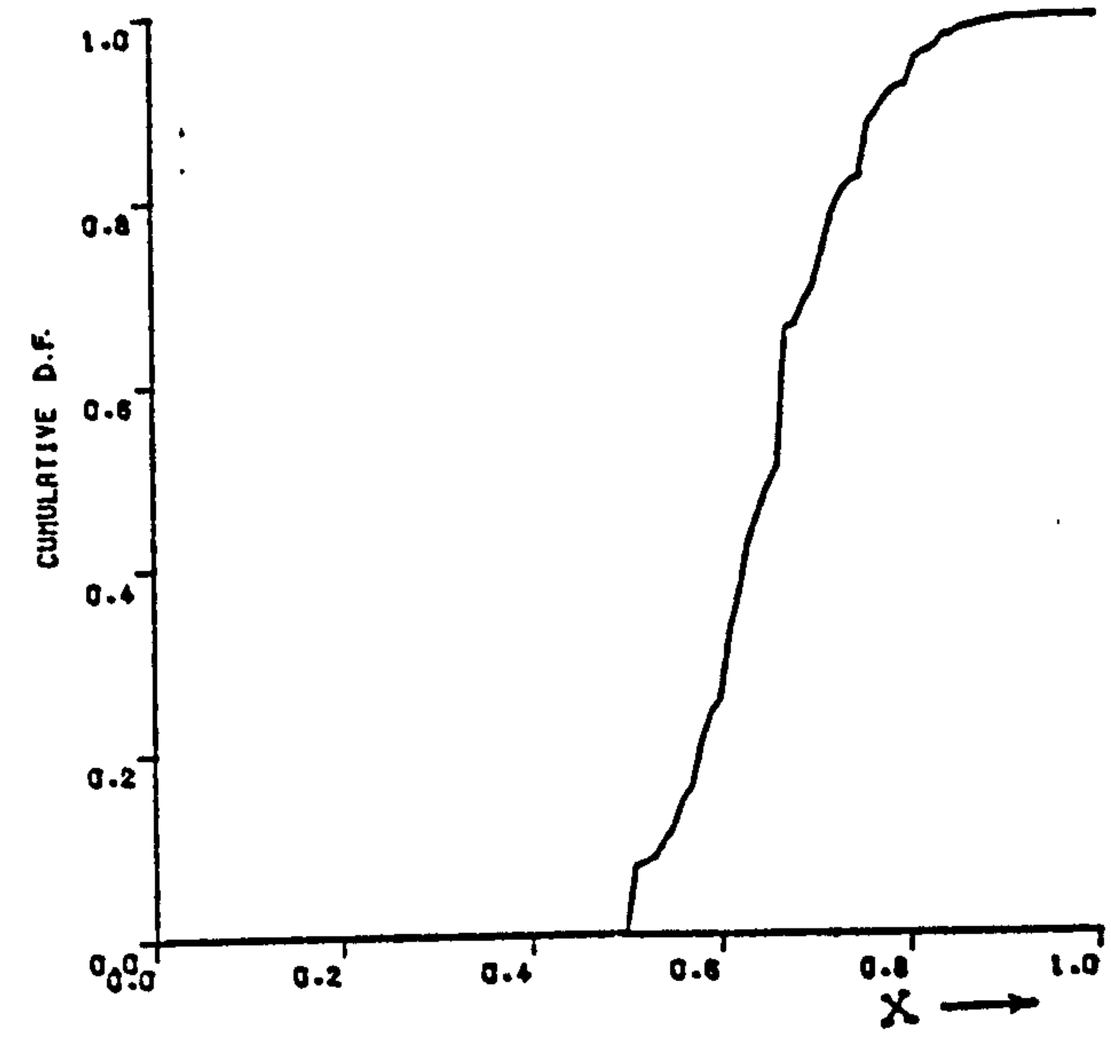
Figure (5.7). Probability of energy clustering in either upper or lower subsets.



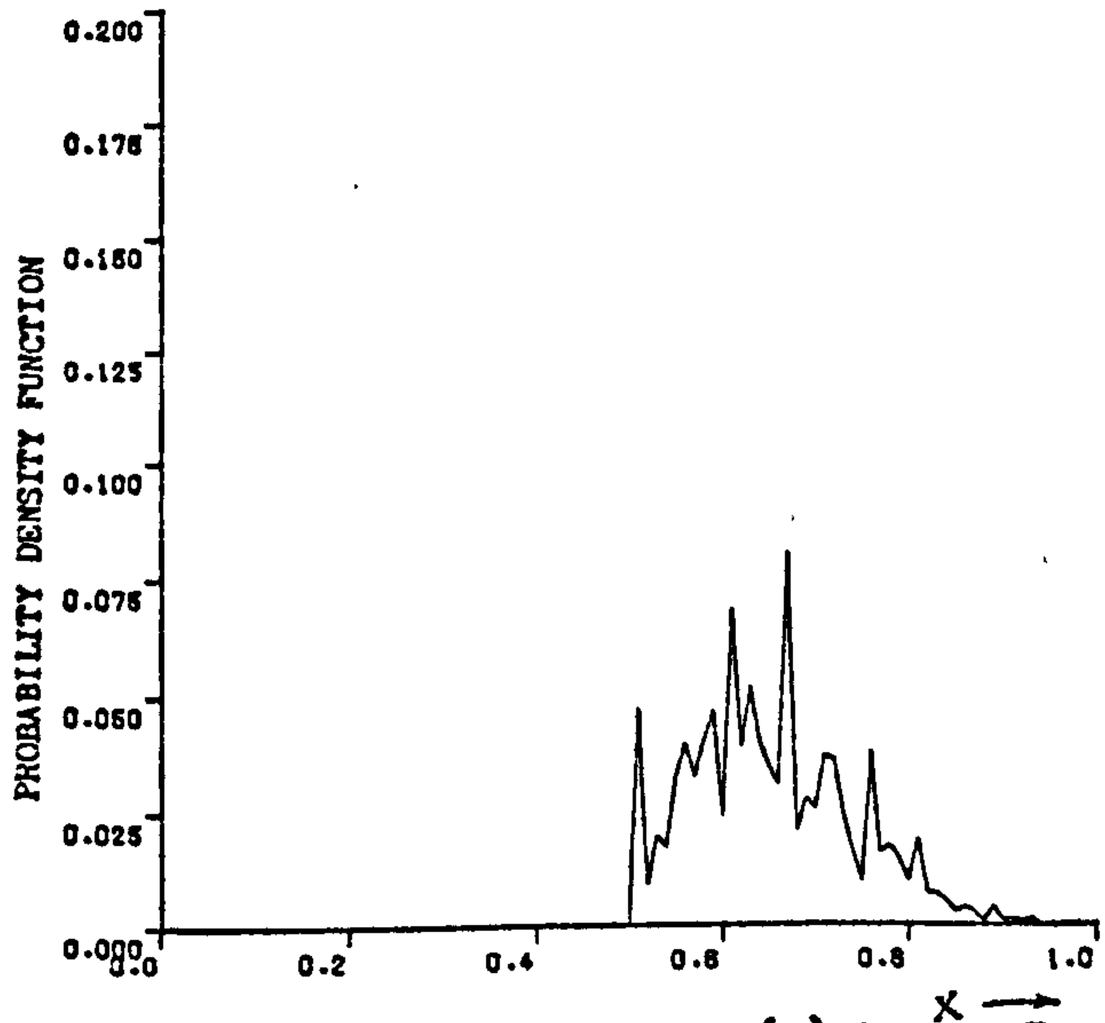


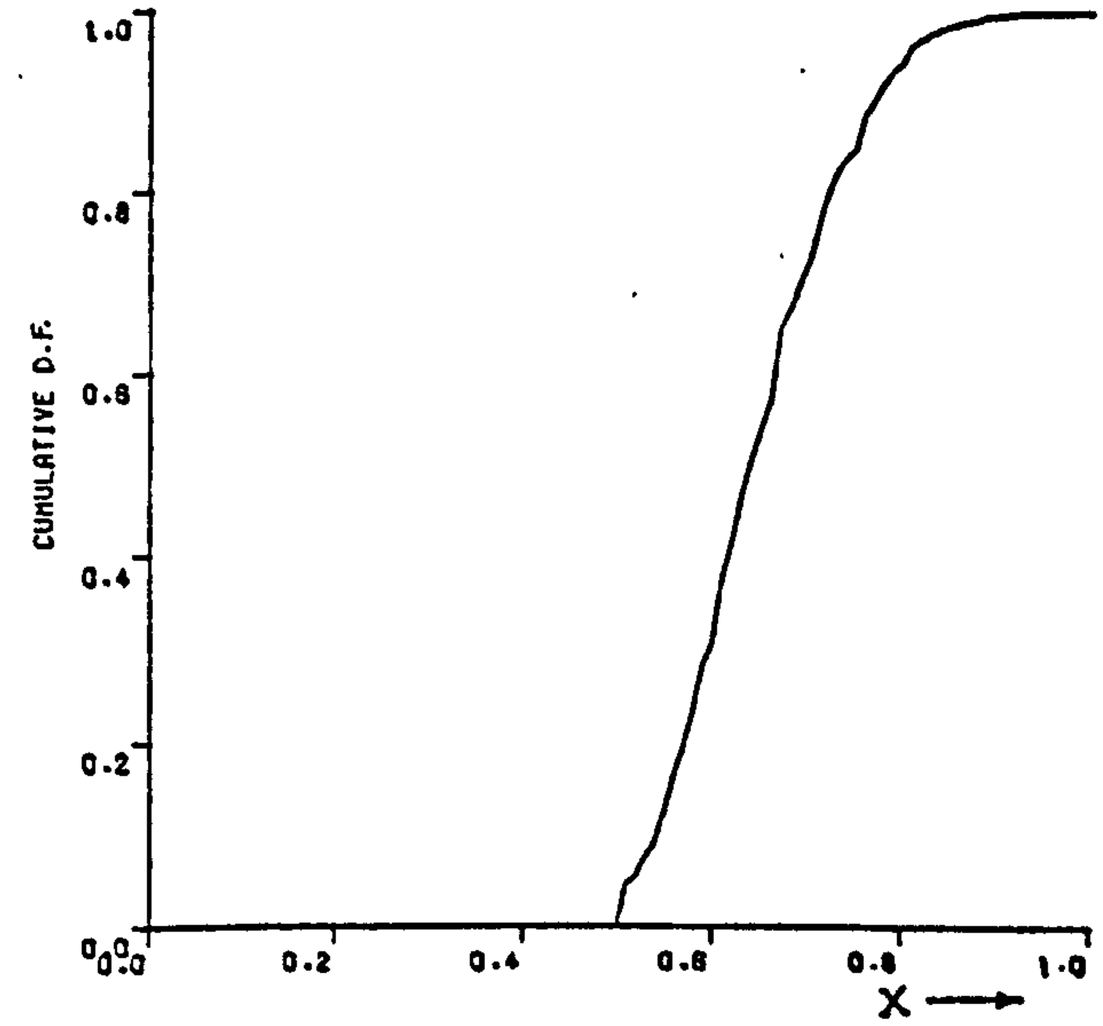
(a) Image A.





(b) Image B.





(c) Image D.

x= proportion of energy content in the most active subset.

Figure (5.8). Probability of energy clustering in any one of four subsets.

(from 0.16 to 0.08), hence reducing the number of cases where the most active subset carries only half (or slightly more) of the total subvector energy.

As in the case of directing index D2, several simulation experiments have been performed to yield the best values of TMIN and TMAX. As a result, the most convenient ranges of values were found to be as follows:

TMIN = 0.5 - 0.6

and TMAX = 2.3 - 2.5.

Computing Index D3 :

Computing Index D3 follows almost the same principles as for the Directing Index D2.

A. Basic Index.

As in case of D2 , Table (5.21) shows the different values of basic index, based on comparing balance B with thresholds TMIN and TMAX.

Table (5.21). Basic index for D3, depending on B.

Comparison result.	Basic index.
$B \leqslant TMIN$	1
$B \geqslant TMAX$	2
$TMIN < B < TMAX$	3

B. Modification index.

In this case, modification index is assigned depending on which subset in the high sequency subvector is the most active one. The following table, Table (5.22), shows the values of such an index for the different possible combinations. Again, as in the case of index D2, the modification index will be added only if the basic index is 3.

Finally, the Directing Index D3 will be :

D3 = Basic index + Modification index (if the basic index is 3).

Table (5.22). Modification index for D3.

The most active subset in high sequency subvector.	Modification index value.
LOWER	0
UPPER	1
ODD	2
EVEN	3

5.3.3.2. Bit Allocation and Complete Coders for D3 :

Table (5.23), shows different combinations of all possible cases concerning Directin Index D3. Number of bits allocated to each one coefficient in each case are also shown, including in this case the dc term H1. It is again to be pointed out that in the case of dc, or near dc vectors, (where no details are present in medium- or high sequency subvectors), the vector is directed to any scheme which codes low sequency coefficients more finely.

Table (5.23). Number of bits allocated for each coefficient in conjunction with D3.

D3	Features of vector. Order :		Number of bits for coefficients : 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
1	B ≤ TMIN		8 4 4 4 3 3 3 3 0 0 0 0 0 0 0 0
2	B ≥ TMAX		8 4 4 4 0 0 0 0 2 2 2 2 2 0 2 0
3	TMIN < B < TMAX	SHLOW =Max.	6 4 3 3 2 2 2 2 2 2 2 2 0 0 0 0
4		SHUP =Max.	6 4 3 3 2 2 2 2 0 0 0 0 2 2 2 2
5		SHODD =Max.	6 4 3 3 2 2 2 2 2 0 2 0 2 0 2 0
6		SHEVEN=Max.	6 4 3 3 2 2 2 2 0 2 0 2 0 2 0 2

Coders to be used with Directing Index D3 are shown, in full, in Tables (5.24 - 5.29).

Table (5.24). Coders for use with Directing Index D3=1.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H4	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H5	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 7
H6	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 7
H7	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 7
H8	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 7
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (5.25). Coders for use with Directing Index D3= 2.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 5 8 11
H3	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H4	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	± 0 2 256
	Representatives	± 0 3
H13	Quant. levels	± 0 2 256
	Representatives	± 0 3
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 2 256
	Representatives	± 0 3
H16	Quant. levels	-
	Representatives	-

Table (5.26). Coders for use with Directing Index D3= 3.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 8 13
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	± 0 3 256
	Representatives	± 1 5
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	± 0 2 256
	Representatives	± 0 3
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 2 256
	Representatives	± 0 3
H16	Quant. levels	-
	Representatives	-

Table (5.27). Coders for use with Directing Index D3= 4.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 3 5 9 256
	Representatives	± 1 3 6 11
H4	Quant. levels	± 0 3 5 9 256
	Representatives	± 1 3 6 11
H5	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	± 0 3 256
	Representatives	± 1 5

Table (5.28). Coders for use with Directing Index D3= 5.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H4	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 6
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	± 0 3 256
	Representatives	± 1 4
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	± 0 3 256
	Representatives	± 1 4
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	± 0 3 256
	Representatives	± 1 4
H16	Quant. levels	-
	Representatives	-

Table (5.29). Coders for use with Directing Index D3= 6.

H2	Quant. levels	± 0 1 2 3 4 6 8 10 256
	Representatives	± 0 1 2 3 4 6 8 11
H3	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 6 256
	Representatives	± 0 2 4 8
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 256
	Representatives	± 1 4
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 4
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 4
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	± 0 3 256
	Representatives	± 1 4

5.4. Assessment Measures :

To assess the advantage of using the stored coders scheme, two different objective measures are introduced here. The first shows the advantage of using multi-coder scheme over the single coder system in the case of non- adaptive coding. The second, measures and assesses the efficiency of directing each subpicture (vector) to its most suitable coder. Anattempt has also been made to reduce the stringency of the objective SNR criteria used, and to take an advantage of noise (error) distribution all over the image area.

5.4.1. Scheme Gain :

This is defined as the increase in the Signal- to -Noise Ratio, (or any other acceptable fidelity criterion), resulting from the use of stored coders (multi-coders) scheme over the use of a single basic coder for all vectors (subpictures) regardless of their different local characteristics. Obviously, the value of this gain will depend on the coder which is used as the general single coder. As it is supposed to deal with all vectors, it is then based on statistical evidence. For the present case, as the fidelity criterion used is the MMSE, a coder based on minimizing this error should be used. Using the appropriate statistical results in the transform domain, for a combination of all images, (Chapter 2), an approximate Max type set of coders is obtained, assuming comparable number of bits for coded coefficients. The resultant coders are shown in Table (5.30).

Figure (5.9) shows a diagram for computer simulation of scheme gain computation.

Table (5.30). Single basic Coding Scheme.

H2	Quant. levels	± 0 2 5 10 17 25 33 43 256
	Representatives	± 0 3 6 12 20 28 37 48
H3	Quant. levels	± 0 4 9 16 256
	Representatives	± 1 6 11 18
H4	Quant. levels	± 0 4 9 16 256
	Representatives	± 1 6 11 18
H5	Quant. levels	± 0 5 256
	Representatives	± 2 7
H6	Quant. levels	± 0 5 256
	Representatives	± 2 7
H7	Quant. levels	± 0 5 256
	Representatives	± 2 7
H8	Quant. levels	± 0 5 256
	Representatives	± 2 7
H9	Quant. levels	± 0 3 256
	Representatives	± 1 5
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	± 0 3 256
	Representatives	± 1 5
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	± 0 3 256
	Representatives	± 1 5
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

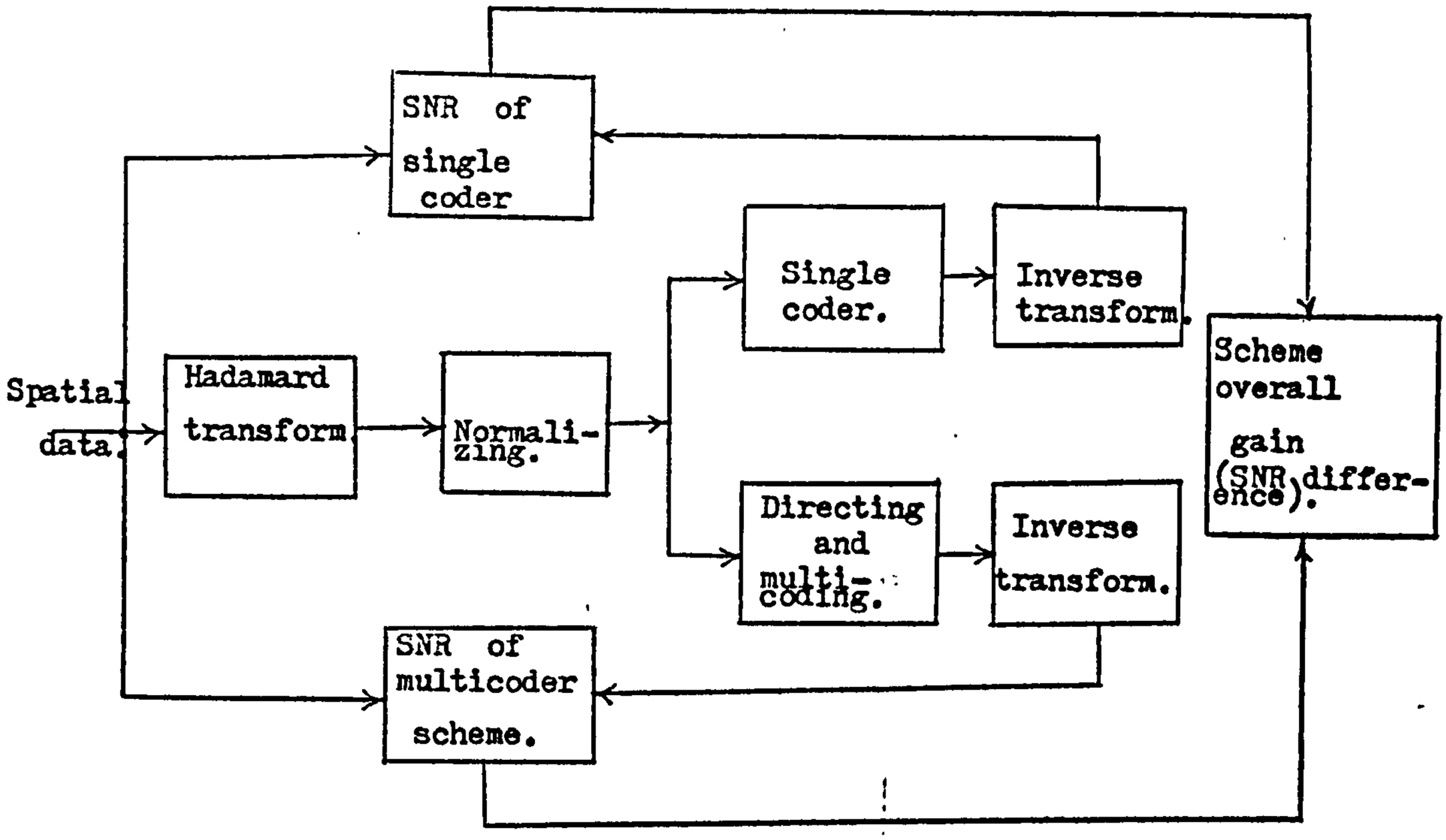


Figure (5.9). Diagram for computation of Scheme overall gain.

Although the total number of bits/vector was 32 in the case of adaptive scheme, the single coder is allowed 4 more bits to be fully comparable with the adaptive one. This is so, because the multicoder scheme will need 'overhead' bits for the identification of directing index, and hence the appropriate coder to be used for a specific vector. For a number of 16 coders, 4 bits will be overheaded for each one subpicture. Although the actual number of coders is considerably less than 16, these 4 bits were considered to allow for any further future expansion.

5.4.2. Directing Index Loss :

This is the second measure in assessing the performance of the whole scheme. It shows the efficiency (or more accurately, the deficiency), by which the proposed directing index does actually direct the vectors (subpictures), to their appropriate coders. In setting such a measure, an entirely hypothetical ideal directing procedure was assumed. This, in fact, is no more than allowing each vector to be coded successively by all available coders. The signal to noise ratio (or any other acceptable

fidelity criterion) is calculted for each individual coder (after decoding and recovering the spatial data). Only the best coder for each vector is considered, and hence, the maximum SNR is computed. The over all maximum SNR is then computed for the whole image. Comparing with the over all SNR obtainable with applying the directing index under test, the drop is then taken as a measure of the suitability of directing index. Obviously, the closer this loss is to zero, the better and more efficient is the directing index.

Figure (5.10) shows a diagram for the computation of this measure. Coders used for the ideal 'free' choice of all vectors, were the same coders used in case of directing index. It should be noted here, that this measure is extremely stringent, as the ideal indexing is entirely hypothetical and is, for the present at least, far from realising

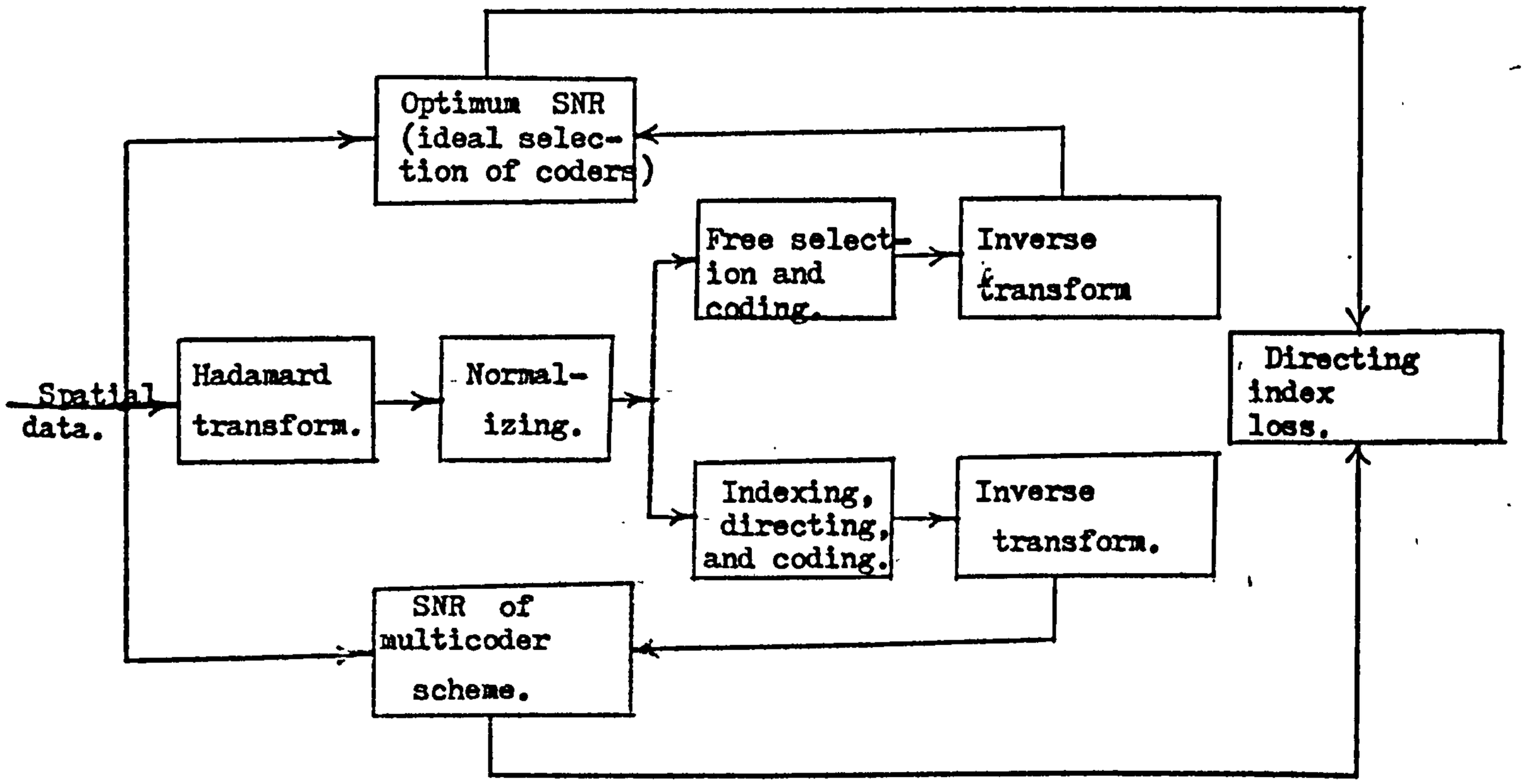


Figure (5.10). Diagram for computation of Directing Index Loss.

5.4.3. ' Exclusive Signal-to-Noise Ratio ':

A major feature associated with objective fidelity criteria is their relative " stringency ". In computing an over all SNR for a whole image, for instance, all pixel (point, sample) errors count, regardless of their actual spatial positions in the image area. A severely noisy reconstructed pixel (or a set of pixels in a vector or block) may well contribute as much noise to the over all SNR as several pixels (or vectors, or blocks), of medium noise level. To illustrate this, let an area be composed of 1000 pixels. Assume 900 of these pixels with a moderate SNR around 10^4. Assume that the remaining 100 pixels, for some reason or another, are very severely distorted and have SNR of about 1 each (i.e. the reconstructed data at each of these pixels was a complete blackout), and that they are randomly distributed all over the area (as the case is likely to be).

Computing an overall SNR as $\sum(\text{noise})^2/(\text{ signal })^2$ will result noise-to-signal ratio =0.1, which will give an SNR of 10. This SNR is 1000 times worse than the average value of most (90%) of the total pixels. This means that only 10 % of the total pixels are responsible for reducing the overall SNR by a factor of 1000. Stringent!

On the other hand, if an attempt is made to compute the SNR directly as $\sum(\text{ signal })^2/(\text{ noise })^2$ (assuming such a computation is realisable), the overall SNR would have been 9000, which is only 1.1 times worse than the average level of SNR in the majority of cases. A disadvantage of such a computation is its impracticality. The computation will not work if the whole area contains any pixel (s) where there is (are) no error at all, (which is quite practical for the majority of cases).

In a subjective assessment, such a low proportion of the total pixels (or vectors) may not contribute any noticeable increase in the overall objectional appearance of the image, especially when, as in the case of Hadamard transform, the noise is always distributed all over the

transform area (vector or block).

In order to get an approximate idea how much the situation may be deteriorated by the presence of such a low proportion of 'irregular' cases (pixels , vectors, or blocks), a modified overall SNR is used in this study. In addition to the different versions of mean-squared-error criterion mentioned and used in Chapter 4, an ' Exclusive ' SNR is introduced here. This criterion is the same as before, except that an arbitrary preset proportion of pixels (vectors or blocks), which have the worst objective measures, are excluded from the computation of the overall signal-to-noise ratio. A value of 10 % was thought to be reasonable and was applied here. This is an entirely arbitrary value, and could be changed in accordance with any formal comparative subjective assessments.

5.5. Results and Comments :

Some results are shown here for testing the different directing indexes devised in this Chapter. Several simulation experiments were performed, involving some variations in coding parameters as well as the thresholds of different coefficients and subvectors in computing these indexes.

Although the ultimate target of any transform coding technique is the reduction of bit rate, in this particular study an arbitrary preset average bit rate of 2 bits/pixel was assumed. This was so chosen to show comparative results with some other indexing adaptive techniques at the same bit rate. Obviously, any increase in the objective fidelity criterion over other adaptive and non-adaptive experiments proves that directing indexes devised here could contributeto the adaptive coding of digital television signals in either of two ways:

1. Improving the performance at the same bit rate of an existing scheme, or
2. Given a satisfactory level of SNR (or any other acceptable fidelity criterion), the average bit rate could be reduced down to the value which gives just this accepta ɔle level.

The major advantage of using the new directing indexes is, however, the absolute independence of the average bit rate from the actual statistical characteristics of subpicture activities. This leads to the constancy of channel bit rate, and eliminates the need for variable coding wordlength which is essential for other techniques which depend on the likelihood of equal distribution of specific activity classes.

Table (5.31) summarizes the results obtained for different tests of adaptive coding using directing indexes. The assessment criteria used were ONSNR and PPSNR (Chapter 4). These were used in both the usual form and the modified ' exclusive ' form, as explained in Section 5.4.3. For quick reference, table numbers of corresponding coders are also given.

Table (5.31). Results of simulating adaptive coding of monochrome signals, using the Directing Indexes,

Directing Index used.	Tables of coders used.	Total Signal-to-Noise ratio.				Gain over the single-coder scheme (Sec. 5.4.1.) in dB.	Loss over an ideal directing scheme (Sec. 5.4.2.)in dB.	Proportion of subpictures having no coding errors at all.
		Usual form.		Exclusive form; Section: 5.4.3.				
		ONSNR dB	PPSNR dB	ONSNR dB	PPSNR dB			
D1	5.4 —— 5.11	33.6	38.1	36.8	40.7	6.6	1.0	42 %
D2	5.15 —— 5.20	34.5	38.9	37.0	41.1	7.5	1.0	45 %
D3	5.24 —— 5.29	35.0	39.3	37.3	41.4	8.0	0.9	48 %

Notes: 1. Results are those obtained for all the test images of Chapter 2.

2. Normalization SNR was about 43.5 dB, ONSNR, or 47.5 dB, PPSNR, (as shown in Chapter 4).

3. SNR for a single-coder non-adaptive coding (Section 5.4.1) was about 27 dB (ONSNR), or 31.5 dB (PPSNR).

4. PPSNR for a monochrome image coding in an experiment with DCT was about 36.6 dB[31], at a bit rate of 2 bits/pixel.

The total SNR shown in the Table is that due to the combined effects of both normalizing, (Chapter 4) and coding. A critical assessment of any coding scheme should exclude the effect of normalization which is constant for the same transform size, and independent from any coding parameters. As shown in Chapter 4, and repeated here for convenience, the SNR due to the sole effect of normalization was about 43.5 dB (ONSNR), or 47.5 dB (PPSNR) for a transform size of N= 16. For more critical assessment of the actual coding efficiency, another measure is included in the table. This is the proportion of the total number of vectors (subpictures)

which have no coding errors at all, and it is seen that this proportion is well over 40 %. The gain indicated in the table is that achieved over the single-coder (non-adaptive) scheme, as mentioned in Section 5.4.1. The loss is the decrease in SNR due to using the Directing Indexes, than the values of SNR would have been obtained using an ideal directing method, as explained in Section 5.4.2.

From Table (5.31) and the notes thereon, it is seen that adaptive coding using the directing indexes achieved an increase in SNR of about (6.5 - 8) dB over the single coder non-adaptive scheme (Section 5.4.1). Compared with an 'ideal' free selection scheme (Section 5.4.2), a slight decrease (loss), of only about 1 dB is shown. This proves the efficiency of the directing indexes.

To show the advantage of using the directing index over the existing scheme, the result of an experiment using a technique similar to the 'activity index'[62] in conjunction with Discrete Cosine Transform[31] is recalled. At an average bit rate of 2 bits/pixel, a PPSNR of about 36.6 dB was reported, as shown in the notes on Table (5.31). Thus, an average improvement of about (1.4 - 2.7) dB (PPSNR) was achieved here, which should be considered as very good, bearing in mind the computational and realisation complexities involved with the Discrete Cosine Transform.

CHAPTER SIX

ADAPTIVE CODING OF COLOUR SIGNALS

The principles of the ' directing index ' devised in Chapter 5 are extended to colour signals. Modifications necessary for different transforms are discussed.

One particular directing index, as an example, is considered in detail, and modified to adapt the colour signals. Both the two values of sampling frequency, namely $3f_{sc}$ and $4f_{sc}$, are studied and the effects on the bit rate are shown. Detailed coders for the different cases of modified directing index are devised and tested subject to the same arbitrary bit rate of 2 bits per pixel as in monochrome.

Results for different cases are also shown, with comparison to monochrome and to other reported tests.

6.1. Introduction :

As a result of investigations carried out in Chapter 3, the effects of subcarrier presence in the colour composite signal could be either exploited, as in the case of $4f_{sc}$ direct transform, or overcome as in the cases of laced- and component transforms. From the results there, it has been shown that both laced and component transforms yield transform domain characteristics which are nearly similar to those of monochrome transform. It has been also shown that these characteristics are virtually independent of sampling frequency. For direct transform, however, significant differences than monochrome transform, as well as great changes between the two sampling frequencies used, were noticed.

In this Chapter, effects of the similarities and changes among the different types of colour transforms and the monochrome one on the adaptive coding of colour signals are considered. The approximate similarity, in the general shape, between the laced and component transforms on one side, and the monochrome transform on the other, suggests that directing indexes devised in Chapter 5 for monochrome adaptive coding could be used for adaptive coding of colour signals in these two types of transforms. This may be subject to some possible slight modifications.

In the case of direct transform, new directing indexes should be necessary due to the differences between characteristics of such a transform and a monochrome one. It should also be more easier to adaptively code $4f_{sc}$ direct transform coefficients than $3f_{sc}$, due to the superior energy compacting characteristics of the first. Principles of directing indexes in selecting the most active groups could still be applied, but after necessary changes.

6.2. Adaptive Coding of Laced Transformed Signals :

As shown in Chapter 3, the general characteristics of the laced transform coefficients are nearly similar to the corresponding coefficients of monochrome line transform, regarding the relative absolute values and variances. Furthermore, this similarity is virtually independent of the sampling frequency. As concluded in Section 3.3, any slight changes, if any, which may be present between the two sampling frequencies $3f_{sc}$, and $4f_{sc}$, do not justify the corresponding increase in the number of coefficients, and the higher bit rate for the case of $4f_{sc}$ sampling frequency. Therefore, only the case of $3f_{sc}$ sampling frequency will be considered here.

Table (6.1) shows the ratios between both the absolute values and the standard deviations of coefficients in different subvectors in an N= 16 laced transformed vector and those of corresponding coefficients in a monochrome line transform. The relative decrease in value of the DC coefficient, H1, and the increase in all the ac coefficients may be explained by the less redundancy of a laced colour signal, compared to a monochrome one, due to the spatial distance of subcarrier interval. This is as expected from discussions in Sections 3.1.2. and 3.2.2.

Table (6.1). Absolute values and standard deviations relative to monochrome values.

Subvector order.	Coeffic.	Ratio of absolute values.	Ratio of standard deviations.
1	1	0.6	0.8
2	2	1.1	0.8
3	3,4	1.1	0.8
4	5 - 8	1.2	0.8
5	9- 16	(1.2 - 1.5)	0.8

Despite these changes in the absolute values of ac coefficients, especially at high sequencies (20% to 50% increase), the average values of relative standard deviations are constant at about 0.8 of the corresponding monochrome transform values. This slight decrease in deviation suggests that no apparent bit reduction can be expected by using the laced transform over the monochrome rates. As the number of bits allocated to each term was roughly based on its standard deviation, a drop of value of deviation of at least 0.5 is required before a single bit could be reduced for any particular coefficient. The increase in absolute values of ac coefficients necessitates the increase of threshold values required for computation of directing indexes in this case. Clearly, the values of quantization levels and representatives of the coders will also change due to the changes in deviations, though this will not be apparent in the first few levels because of the nature of integer values concerned.

Using these values of both absolute values and standard deviations of coefficients, the directing indexes D1, D2, and D3 could be directly computed and used for adaptive coding of laced transformed signals in a similar manner as in monochrome signals.

6.3. Adaptive Coding of Component Transformed Signals :

As the value of a luminance component Y in a composite colour signal is always positive and normally larger than both of the chrominance components U, and V, some differences are expected in the characteristics of their corresponding transform domains. As shown in Sections 3.1.3. and 3.2.3., value of H1 in a chrominance transform may have positive and negative values depending on the colour characteristics of a local area. In addition to that, the absolute values of chrominance transform ac coefficients are relatively low. Although both u and v components may have values widely different from each other, the average absolute values of their corresponding transform coefficients are, on statistical bases, nearly the same.

Table (6.2) shows the ratios between both absolute values and standard deviations of coefficients in a luminance component Y-transform and those of corresponding coefficients in a monochrome line transform for a vector size of N= 16. The table shows that coefficient values and standard deviations are always less than the corresponding monochrome. values. It also shows that the general trends of this decrease for both averages and deviations are nearly similar. Relative deviations of coefficients in the low and medium sequency subvectors drop by more than a half, while in the high sequency subvector the relative values are almost one fourth. This suggests that coefficients in low and medium sequency subvectors could be coded with one bit fewer coding bits than in monochrome, while for high sequency subvector, coefficients may be coded with two bits fewer. For the DC term H1, however, no apparent saving is possible. The same may well apply to H2, but in this case a more compressed coder may be used.

Table (6.2). Absolute values and standard deviations of Y-transform coefficients relative to monochrome.

Subvector order.	Coeffs.	Relative value of absolute average.	Relative value of standard deviation.
1	1	0.65	0.75
2	2	0.6	0.6
3	3,4	0.5	0.4
4	5 - 8	(.33 - .5)	(.25 -.35)
5	9 - 16	.25	.25

Table (6.3). Absolute values and standard deviations of U- and V- transforms relative to monochrome values.

Subvector order.	Coeffs.	Relative value of absolute average.	Relative value of standard deviation.
1	1	.08	.3
2	2	.25	.2
3	3,4	.25	.2
4	5 - 8	.3	.15
5	9 - 16	.25	.15

Table (6.3) shows the same characteristics for the chrominance component - transforms. The first feature to be noted from Table (6.3) is the very low value of H1. In spite of that, its standard deviation is relatively high, (compared with its relative absolute value), being about a third of the corresponding monochrome transform value. Standard deviation and absolute values of other ac coefficients are considerably lower than that of monochrome transform. This will allow, theoretically at least, substantial reductions in coding bits needed for these coefficients.

As an example of the probable levels of bit rate in component-transforms, let us take the case of a 16- coefficient vector of a relatively quiet activity, which would have been categorized as Directing Index D1=1, in Chapter 5. From Table (5.3.), and taking the corresponding drop in values of standard deviations of different coefficients in cosideration, Table (6.4) shows the appropriate number of bits which will be necessary for all the three components. Original number of bits for monochrome coefficients are repeated here for convenience and quick reference. Where two values are shown for some coefficients, the smaller is for the minimum number of bits, and the bigger is for better coding.

Table (6.4). Bit allocation in Components transform, for Index 1.

Coeff.	Bits for monoch.	Y- component.		U or V component.	
		Rel.Dev.	No. of bits	Rel. Dev.	No. of bits.
1	8	.75	8	.3	7- 6
2	3	.6	3 - 2	.2	2
3	3	.4	2	.2	2
4	3	.4	2	.2	2
5	3	.35	2	.18	0 or 2
6	2	.31	0 or 2	.18	
7	2	.27		.16	
8	2	.25		.15	
9	2	.25		.15	0
10	2	.25		.15	0
11	2	.22		.15	0

From the Table, the minimum number of bits for the whole vector is:

16 (for Y) + 12 (for U) + 12 (for V) = 40

which means an average rate of 2.5 bits /pixel. Although this seems to be about 25% higher than in the monochrome bit rate, it is to be noted that

performance is expected to fall well behind corresponding monochrome level due to more coefficients been discarded. If a more faithful coding is to be achieved, then by adding the bits required for coding some of the left coefficients, the total number of bits will be increased to about 60 or more, with an average rate of more than 3.75 bits/pixel, for a performance nearly comparable with monochrome.Table (6.5) shows the SNR criteria for a Y- component transform coding, using the Directing Indexes D1, D2, and D3 subject to an average bit rate of 2 bits/pixel. Values for both chrominance transforms at additional 2 bits/pixel were nearly in the same ranges of values.

Table (6.5). SNR's of Y- component transform coding using Directing Indexes D's subject to a rate of 2bits/pixel.

Directing Index.	$3f_{sc}$		$4f_{sc}$	
	ONSNR dB	PPSNR dB	ONSNR dB	PPSNR dB
D1	34.45	42.54	36.10	44.27
D2	35.50	43.63	37.63	45.65
D3	34.95	43.06	36.35	44.43

Although there is about (1.5 - 2) dB increase in case of $4f_{sc}$ sampling frequency over the $3f_{sc}$, it should be borne in mind that the actual bit rate at the $4f_{sc}$ is in fact 4/3 times the normal bit rate, because of the increased number of coefficients. In general, there is about 4 dB increase over that of corresponding monochrome signals. This is due to the decrease in dynamic ranges and standard deviations of coefficients which lead to more fine coding (compressed coders than in case of monochrome). However, if a comparable performance is required from chrominance component transforms as well, the over all bit rate will be in the range of 4 bits/pixel (i.e. double that of monochrome), as shown in Table (6.4).

As a result, a conclusion could be reached that Directing Indexes devised in Chapter 5 for the monochrome signals adaptive coding could be used in conjunction with both laced and component transforms of colour signals. The principles of indexing will need not be changed, apart from the values of thresholds of different subvectors and the balance measures. Coders should be changed in order to allow for the changes in standard deviations, especially for the case of chrominance component transforms.

6.4. Adaptive Coding of $3f_{sc}$ Direct Transformed Signals :

Transform domain characteristics of a direct transform at $3f_{sc}$ sampled signals were discussed in Sections 3.1.1.1. and 3.1.1.2. It was shown there, that ac energy is spread over more coefficients in the transform domain than in the case of monochrome transform. This fact will lead either to inferior performance if the number of coefficients and number of coding bits were limited, or to an increase in bit rate if a reasonable performance is to be achieved.

From Figures (3.2. and 3.4.), Chapter 3, and from other statistical studies, it has been found that, although some coefficients seem to have always relatively high energy content, like H6, and H11, some others may vary considerably, from some subpictures to others, in their ranges of values (and hence energy content), as the case with H5, H13, and H14. For a limited number of coding bits available for a subpicture, those coefficients which have always, or in the great majority of cases, the bulk of energy content, will be unconditionally coded. Any coding bits left after that will be adaptively allocated to some of the remaining coefficients, subject to a suitable means of directing.

Directing indexes devised in Chapter 5 could be modified for application in the present case. In this Section, Index D1 will be modified into another index, C1, for use with the direct transform at $3f_{sc}$ sampling frequency. Figure (6.1) shows a 16- coefficient vector with two groups of coefficients. The first group contains those coefficients which have, in general, most of the energy, while the second contains the remaining coefficients. Dividing the vector again into its basic subvectors and naming them as in Chapter 5, corresponding activity measures could be similarly computed. In this case, activity measure for a particular subvector will be taken as the sum of absolute values of its coefficients, excluding those belonging to group 1, as they are always coded.

Procedures for computing the directing index C1 are similar to those of D1, except for the following changes :

$$A_{medium} = \sum_{i=5}^{8} |Hi| - |H6| \quad \ldots\ldots\ldots\ldots\ldots\ldots (6.1).$$

$$A_{high} = |H9| + \sum_{i=13}^{16} |Hi| \quad \ldots\ldots\ldots\ldots\ldots\ldots (6.2).$$

where A_{medium}, and A_{high} are the activity measures for medium and high sequency subvectors, respectively. Obviously, the values of thresholds for medium and high sequency subvectors should be reduced due to the exclusion of coefficients with high values. Table (6.5) shows the coding bits allocated to different coefficients for different values of directing index C1.

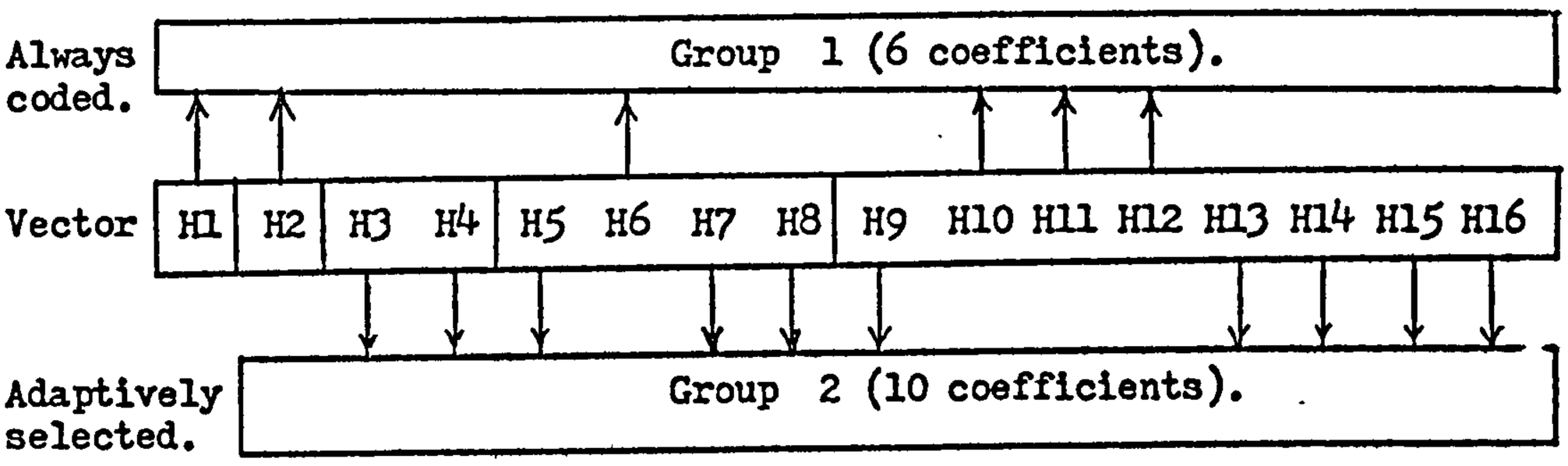


Figure (6.1). Direct transform vector at $3f_{sc}$, and its groups.

Coders to be used in association with directing index C1 are shown, in full, in Tables (6.6. - 6.13.).

The set of thresholds used was 5, 8, and 7.

Table (6.5). Bit allocation for different values of directing index C1, subject to a bit rate of 2 bits/pixel.

C1	Bits for coefficients: H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16
1	8	3	0	0	0	3	2	0	2	3	4	3	2	2	0	0
2	8	3	0	0	0	3	0	0	2	3	4	3	2	2	2	0
3	8	3	0	0	2	3	2	0	2	3	4	3	2	0	0	0
4	8	3	0	0	2	3	2	0	0	3	4	3	2	2	0	0
5	8	3	3	3	2	3	0	0	0	3	4	3	0	0	0	0
6	8	3	2	2	0	3	0	0	0	3	4	3	2	2	0	0
7	8	3	2	2	2	3	2	0	0	3	4	3	0	0	0	0
8	8	3	2	2	0	3	0	0	2	3	4	3	2	0	0	0

Table (6.6). Coders to be used with Directing Index Cl= 1.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	-
	Representatives	-
H4	Quant. levels	-
	Representatives	-
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 5 256
	Representatives	± 2 8
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 16 21 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	± 0 5 256
	Representatives	± 2 8
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.7). Coders to be used with Directing Index C1= 2.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	-
	Representatives	-
H4	Quant. levels	-
	Representatives	-
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 5 256
	Representatives	± 2 8
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 16 21 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	± 0 5 256
	Representatives	± 2 8
H15	Quant. levels	± 0 5 256
	Representatives	± 2 8
H16	Quant. levels	-
	Representatives	-

Table (6.8). Coders to be used with Directing Index C1= 3.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	-
	Representatives	-
H4	Quant. levels	-
	Representatives	-
H5	Quant. levels	± 0 5 256
	Representatives	± 2 8
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 5 256
	Representatives	± 2 8
H10	Quant. levels	± 0 4 7 12 256
	Representatives	± 1 5 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 16 21 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.9). Coders to be used with Directing Index C1= 4.

H2	Quant. levels	± 0 4 8 14 . 256
	Representatives	± 1 5 10 16
H3	Quant. levels	-
	Representatives	-
H4	Quant. levels	-
	Representatives	-
H5	Quant. levels	± 0 5 256
	Representatives	± 2 8
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 15 20 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	± 0 5 256
	Representatives	± 2 8
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.10). Coders to be used with Directing Index Cl= 5.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 8 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 8 13
H5	Quant. levels	± 0 5 256
	Representatives	± 2 8
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 , 256
	Representatives	± 0 3 6 9 12 15 20 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.11). Coders to be used with Directing Index Cl= 6.

H2	Quant. levels	± 0 4 8 14 . 256
	Representatives	± 1 5 10 16
H3	Quant. levels	± 0 5 256
	Representatives	± 2 8
H4	Quant. levels	± 0 5 256
	Representatives	± 2 8
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 15 20 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	± 0 5 256
	Representatives	± 2 8
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.12). Coders to be used with Directing Index Cl= 7.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	± 0 5 256
	Representatives	± 2 8
H4	Quant. levels	± 0 5 256
	Representatives	± 2 8
H5	Quant. levels	± 0 5 256
	Representatives	± 2 8
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 15 20 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.13). Coders to be used with Directing Index C1= 8.

H2	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 16
H3	Quant. levels	± 0 5 256
	Representatives	± 2 8
H4	Quant. levels	± 0 5 256
	Representatives	± 2 8
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 5 256
	Representatives	± 2 8
H10	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H11	Quant. levels	± 0 2 5 8 11 14 18 24 256
	Representatives	± 0 3 6 9 12 16 21 27
H12	Quant. levels	± 0 3 7 12 256
	Representatives	± 1 4 9 15
H13	Quant. levels	± 0 5 256
	Representatives	± 2 8
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

6.5. Simple Coding of $4f_{sc}$ Sampled Direct Transformed Signals :

As seen in Section 3.2.1.2. (in Chapter 3), most of the ac energy in a $4f_{sc}$ sampled direct transformed signal of a transform size N, is associated with the two mid-sequency terms, in the transform domain, of orders N/2 and N/2 +1. It has been suggested there, that a possible low rate coding scheme may be to code these two ac coefficients in addition to the first (dc) term, H1. This ,of course, will greatly reduce the average bit rate especially for larger vector sizes like N= 64, or 128.

From Figure (3.16), Chapter 3, and according to the values of relative variances of the mid-sequency coefficients, these coefficients could be coded using from (1 to 3) bits fewer than coding bits of the dc term. Assuming a 6-bit coder for each of these coefficients, to allow more fine coding, a total of 20 bits will, basically, be sufficient to code a subpicture in such a scheme.This is equivalent to about 26.7 bits (compared with $3f_{sc}$ and monochrome number of coefficients), leading to an average bit rate of 1.67, 0.83, and 0.42 bits/pixel for vector sizes N= 16, 32, and 64 respectively.

This simple low rate scheme (which will be called here 3-coefficient scheme,for simplicity), has been tested, and the results are shown in Table (6.14).

Table (6.14). SNR's of the 3-coefficient simple coding scheme.

Vector size N.	ONSNR dB.	PPSNR dB.	Effective bit rate.
16	24.97	32.92	1.67
32	20.39	28.41	0.83
64	18.05	26.11	0.42

The decrease in fidelity as the vector size increases, could be explained by the fact that, at larger vector sizes, more and more energy is spread to other low and medium sequency coefficients away from the

mid-sequency coefficients (Section 3.2.1.2). Obviously, the very low values of SNR's at vector sizes larger than 16 may prohibit such a scheme from being used without modification.

A direct and logical modification to the previous 3-coefficient scheme is to allow some other coefficients, which have a considerable proportion of the ac energy not contained in the mid-sequency terms, to be coded in addition to the three coefficients. From Section 3.2.1.2, it seems that most of this energy is clustered in the lowest 1/8 th sequency coefficients. This is clear from the constancy of energy packing efficiency in the region (1/8 - 1/2) of total coefficients, Figure (3.17). Then, allowing coefficients (H2 to $H_{N/8}$) to be coded, in addition to the other mentioned in the 3-coefficient scheme, will increase the fidelity of the recovered data. Inevitably, this will add to the average bit rate.

Considering a 5-bit coder for H2, and an average 4-bit coder for each of the other coded low sequency coefficients (as the relative values of variances and deviations suggest), Table (6.15) shows a summary of bit rate in each case of vector sizes. The proportion of ac energy not contained in the mid-sequenc' terms is also shown for each vector size.

Table (6.15). Average bit rate for N/8 coding scheme.

Size N	Mid-seq. coeff.order	AC energy proportion not in mid.	Other coded coeffs.	Total no. of bits.	Apparent bit rate.	Effective bit rate.
16	8,9	20 %	H2	25	1.563	2.08
32	16,17	32 %	H2-H4	33	1.03	1.375
64	32,33	45 %	H2-H8	49	0.766	1.02
128	64,65	63 %	H2-H16	81	0.633	0.844

For simplicity, this scheme will be called N/8 coding scheme, although the actual number of coded coefficients is N/8 +3.

Testing this scheme , much better results for performance than in the case of 3-coefficient scheme were obtained as shown in Table (6.16). Comparing Table (6.16) and Table (6.14), a substantial increase in SNR of about 6 dB is achêived with a very little increase in average bit rate (about 25 % in the case of N= 16). This proves the property of the extreme suitability of the $4\ f_{sc}$ sampling frequency for the Hadamard transform applications.

Table (6.16). SNR's for N/8 coding scheme.

Size N.	ONSNR dB.	PPSNR dB.	Actual rate.
16	28.34	36.53	2.08
32	26.29	34.92	1.375
64	25.26	33.85	1.02

Comparing Table (6.16) with results of monochrome transform shows that for a vector size of N= 16, the $4f_{sc}$ direct transform at an actual rate of about 2 bits/pixel is better than monochrome transform at an actual rate of 2.25 bits/pixel (including the bits required for directing). For vector larger than 16, an approximate idea about fidelity could be obtained from Table (6.16), bearing in mind, as a result of Chapter 4, that the decrease in SNR includes an average of about 2.5 dB per each bit increase in the binary word expressing the vector size (i.e. each time the vector size is doubled), due to normalizing alone.

6.6. Adaptive Coding of $4f_{sc}$ Sampled Direct Transform Signals :

Like the case of $3f_{sc}$ sampled direct transform, an adaptive coding scheme could be devised in which the coefficients which always have high dynamic ranges will escape indexing and will be unconditionally coded. Remaining available coding bits will be adaptively allocated to some coefficients which have more energy than others. For illustration, the case of a vector size of N= 16, with the simplest index D1 (of Chapter 5) will be considered. This index could be modified to a new one, C2 in this case, for use with $4f_{sc}$ direct transform coding. Unlike the case of $3f_{sc}$, coefficients to escape indexing in a $4f_{sc}$ sampled direct transform are few and are easily located as H2, H8, and H9 (H2, $H_{N/2}$, $H_{N/2+1}$ in the general case of a vector size N). Thus, a modified index C2 could be based on the modified subvector activity measures which will then be taken as follow :

$$A_{low} = |H3| + |H4| \quad \ldots\ldots\ldots\ldots\ldots (6.3).$$

$$A_{medium} = \sum_{i=5}^{i=7} |Hi| \quad \ldots\ldots\ldots\ldots\ldots (6.4).$$

$$A_{high} = \sum_{i=10}^{i=16} |Hi| \quad \ldots\ldots\ldots\ldots\ldots (6.5).$$

As the largest coefficient in each of the medium and high sequency subvector is excluded from each subvector activity measure, the corresponding thresholds should also be considerably reduced.

Table (6.17) shows bit allocation for each of the coefficients for different values of Directing Index C2, subject to an 'apparent' bit rate of 2 bits/pixel. Although the number of bits allocated in schemes 3, 7, and 8 are identical, other coding parameters (quantizing levels and representative-values) were varied subject to the studied characteristics.

Although, as mentioned before, coefficients H2, H8, and H9 will all escape the process of indexing, they will be , still, nonlinearly coded due to the limited number of bits allocated to each of them.

Table (6.17). Bit allocation for different values of C2, subject to an apparent bit rate of 2 bits/pixel.

C2	Bits allocated for coefficients: H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16
1	8	4	2	2	2	2	2	4	4	2	0	0	0	0	0	0
2	8	4	2	2	0	2	0	4	4	2	0	2	0	2	0	0
3	8	4	3	3	2	2	2	4	4	0	0	0	0	0	0	0
4	8	4	2	2	2	2	2	4	4	2	0	0	0	0	0	0
5	8	4	3	3	2	2	2	4	4	0	0	0	0	0	0	0
6	8	4	3	3	0	0	0	4	4	2	0	2	0	2	0	0
7	8	4	3	3	2	2	2	4	4	0	0	0	0	0	0	0
8	8	4	3	3	2	2	2	4	4	0	0	0	0	0	0	0

Coders to be used in conjunction with Direct Index C2 are shown, in full, in Tables (6.18 - 6.25).

The set of thresholds used was 6, 4, and 4.

Table (6.18). Coders to be used with Directing Index C2= 1.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.19). Coders to be used with Directing Index C2= 2.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.20). Coders to be used with Directing Index C2= 3.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H4	Quant. levels	± 0 2 4 7 256
	Representatives	± 0 2 5 9
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.21). Coders to be used with Directing Index C2= 4.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.22). Coders to be used with Directing Index C2= 5.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.23). Coders to be used with Directing Index C2= 6.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	± 0 3 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	± 0 3 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	± 0 3 256
	Representatives	± 1 5
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.24). Coders to be used with Directing Index C2= 7.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.25). Coders to be used with Directing Index C2= 8.

H2	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H9	Quant. levels	± 0 3 6 9 12 16 20 26 256
	Representatives	± 1 4 7 10 13 17 22 28
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

6.7. Adaptive Coding of $4f_{sc}$ Direct Transform with Actual Bit rate of 2 Bits/pixel :

Although coders used in previous Section were devised subject to an average bit rate of 2 bits/pixel, the actual rate was about 2.7, due to the increased number of coefficients over the $3f_{sc}$ sampled colour signals or the monochrome transform considered in Chapter 5. For a fully comparable coding bit rate, the average bit rate in Section (6. 6.) should be reduced. For a vector size of N= 16, a total of 24 bits will lead to an "actual" bit rate of 2 bits/pixel. In order to achieve such a reduced rate, some ways of bit conservation are necessary. First, the DC term is coded with a 6- bit linear coder instead of the 8- bit used so far. Secondly, number of bits allocated for high value coefficients are reduced and hence their coders are relatively expanded.

Table (6.26) summarizes the number of bits allocated to different coefficients subject to an 'actual' bit rate of 2 bits/pixel (24 bits for the whole vector), for the same Directing Index C2.

Coders to be used with Directing Index C2 subject to the new bit rate mentioned here are shown, in full, in Tables (6.27 - 6.34).

Table (6.26). Bit allocation for Directing Index C2, subject to an 'actual' bit rate of 2 bits/pixel.

C2	Bits allocated for coefficients :															
	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16
1	6	2	2	2	2	2	2	3	3	0	0	0	0	0	0	0
2	6	4	2	2	0	2	0	4	4	0	0	0	0	0	0	0
3	6	2	2	2	2	2	2	3	3	0	0	0	0	0	0	0
4	6	4	2	2	2	2	0	3	3	0	0	0	0	0	0	0
5	6	4	3	3	2	0	0	3	3	0	0	0	0	0	0	0
6	6	4	3	3	0	0	0	4	4	0	0	0	0	0	0	0
7	6	2	2	2	2	2	2	3	3	0	0	0	0	0	0	0
8	6	2	2	2	2	2	2	3	3	0	0	0	0	0	0	0

Table (6.27). Coders to be used with Directing Index C2=1.

H2	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H1C	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.28). Coders to be used with Directing Index C2= 2.

H2	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H9	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.29). Coders to be used with Directing Index C2=3.

H2	Quant. levels	± 0 6 256
	Representatives	± 2 8
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.30). Coders to be used with Directing Index C2=4.

H2	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H3	Quant. levels	± 0 4 256
	Representatives	± 1 6
H4	Quant. levels	± 0 4 256
	Representatives	± 1 6
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.31). Coders to be used with Directing Index C2= 5.

H2	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 17 22
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.32). Coders to be used with Directing Index C2= 6.

H2	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H3	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H4	Quant. levels	± 0 3 6 10 256
	Representatives	± 1 4 7 13
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H9	Quant. levels	± 0 2 4 6 9 12 15 19 256
	Representatives	± 0 2 4 7 10 13 16 22
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.33). Coders to be used with Directing Index C2= 7.

H2	Quant. levels	± 0 6 256
	Representatives	± 2 8
H3	Quant. levels	± 0 6 256
	Representatives	± 2 8
H4	Quant. levels	± 0 6 256
	Representatives	± 2 8
H5	Quant. levels	± 0 4 256
	Representatives	± 2 6
H6	Quant. levels	± 0 4 256
	Representatives	± 2 6
H7	Quant. levels	± 0 4 256
	Representatives	± 2 6
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (6.34). Coders to be used with Directing Index C2= 8.

H2	Quant. levels	± 0 6 256
	Representatives	± 2 8
H3	Quant. levels	± 0 6 256
	Representatives	± 2 8
H4	Quant. levels	± 0 6 256
	Representatives	± 2 8
H5	Quant. levels	± 0 4 256
	Representatives	± 1 6
H6	Quant. levels	± 0 4 256
	Representatives	± 1 6
H7	Quant. levels	± 0 4 256
	Representatives	± 1 6
H8	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H9	Quant. levels	± 0 4 8 14 256
	Representatives	± 1 5 10 17
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

6.8. Results and Comments :

Table (6.35) summarizes the simulation results of adaptive coding using the directing indexes discussed in this Chapter, and applied to all colour test images of Chapter 3. For simplicity, and direct comparison with other works, only the usual PPSNR criterion is used here (i.e. without exclusion).

Table (6.35). Simulation results for different cases of colour signals adaptive coding.

Directing index.	Sampling freq.	Coders'tables.	Total PPSNR in dB.	Average bit rate
C1	$3f_{sc}$	6.6 - 6.13	30.1	2 bits
C2	$4f_{sc}$	6.18 - 6.25	41.8	2 bits apparent.
C2	$4f_{sc}$	6.27 - 6.34	39.5	2 bits actual.

The table shows the extreme superiority of direct transform adaptive coding at $4f_{sc}$ over that at $3f_{sc}$. For an 'apparent' bit rate of 2 bits/pixel an increase of 11.7 dB is possible, while at an'actual' rate of 2 bits/pixel (i.e. taking the actual number of pixels into account), an improvement of more than 9 dB is still noticed. Comparing with results of Section 6.5, the performance of the f_{sc} ' simple ' direct coding at a lower rate of 1.67 bits is still better than of $3f_{sc}$ adaptive coding at 2 bits/pixel.

As for other directing tests, a PPSNR of a luminance component(Y) coding of about 40 dB at an average bit rate of 2 bits/pixel has been reported[31]. The directing criterion used was similar to the activity index[62], in conjunction with the Discrete Cosine Transform. The image used was of a single category of images, namely a portrait, where the Y-component is expected to be relatively quiet and less active than other normal images.

Directing indexes D2 and D3, (Chapter 5), could be modified for use with colour adaptive coding, as was the case with D1. This will involve the study of energy distribution in transform domain. Following nearly similar procedures as in Sections 5.3.2. and 5.3.3, the two new directing indexes could readily be developed, together with corresponding sets of coders.

CHAPTER SEVEN

DOUBLE RANGE CODERS

Conditional characteristics of different coefficients and groups of coefficients are studied. A special coder, called here as ' Double Range Coder ', is used to achieve better performance. In this coder, the range of the lowest sequency ac coefficient is divided into two ranges, low and high, each having its own coder. As a result of conditional characteristics, one of two different sets of coders for other ac coefficients will be associated with either of these two coders. These sets will be referred to as 'Compressed-' and 'Expanded'-range coders, respectively. The different coders are devised and tested, with the improvements in performance pointed out.

Although this Chapter is restricted to monochrome line transform, its principles could easily be extended to block transformation as well as to colour signal transforms.

7.1. Introduction :

One of the major sources of quantizing errors in coding normal subpictures, is the limited number of coding bits/coefficient available. In the examples shown here so far, the number of bits used for coding any ac coefficient was restricted to a maximum of 4 bits. To allow more accurate coding for high dynamic range coefficients, (say H2), the use of larger quantizing steps is therefore inevitable. This will, as known, reduce the SNR of the recovered data. On the other hand, any attempt to improve the SNR by allocating more bits for high dynamic range coefficients will increase the average bit rate of the transmitting channel. Use of the stored coders scheme, however, can introduce some improvements in SNR while still employing the same average bit rate. This is achieved by the use of a special coding scheme, which will be called ' Double Range Coder', for coding high dynamic range coefficients.

In order to accurately code the low values of a high dynamic range coefficient,(which represent the majority of cases), a separate coder is assigned for them. This coder will code only the values which are equal to, or less than, a preset threshold value. In the rest of cases, which are generally few in proportion, another coder is used, which considers the next higher level, to be its lowest level. The upper level of this second coder will be, of course, the highest value which the corresponding coefficient can have.

To illustrate, let the dynamic range of H2 be (0 - 50), on a normalized (0 - 255) scale. Assigning a threshold of 7, for simplicity, to direct the vector (subpicture) to either of two different coders A or B, each of 4 bits. Coder A which deals with the coefficient whenever its absolute value is equal to or less than 7, will be, in this case, an excellent coder for that range, due to its sufficient levels. Thus, a considerable improvement in performance is expected. The second coder, B

will deal with the same coefficient whenever its absolute value is larger than the threshold 7, i.e. in the range (8 - 50). Considering the level of 8 as its starting level, the coder's quantizing steps will be smaller than in the case of an ordinary 'full range coder',,hence, another improvement in SNR can be expected. Unfortunately, number of coders will be doubled each time this principle is applied for any one of the high dynamic range coefficients.

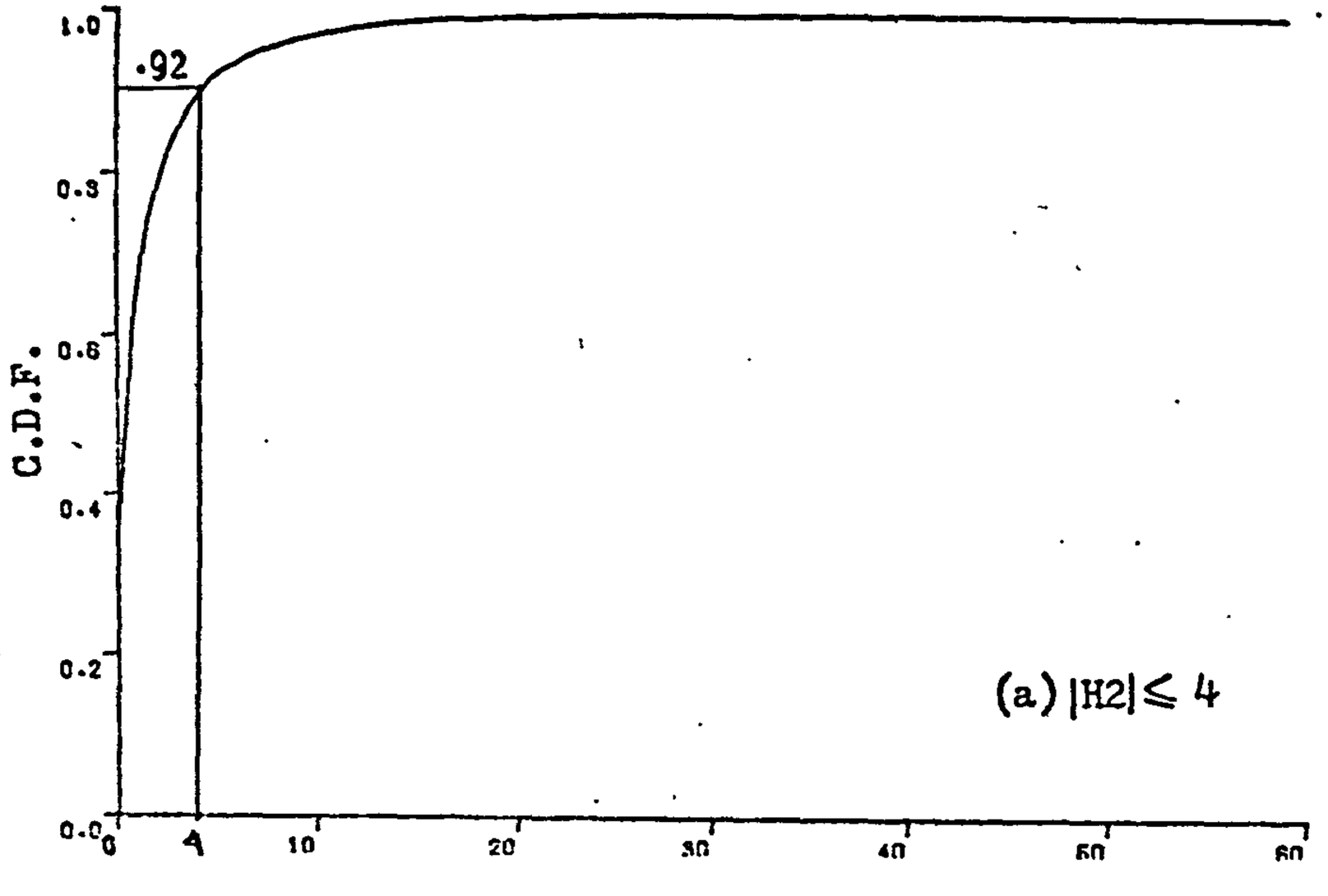
Thinking of the coefficients in the low sequency subvector, H3 and H4, as well as the coefficient H2 as the ones having the highest dynamic ranges, this will require 8 times the ordinary number of coders, which is too large to be practical. Neverthless, careful consideration of statistical evidence shows an important fact. It is highly unlikely that the range of values of H3 and H4 will differ too much from the range of values of H2 . In other words, it is more likely that the value of both H3 and H4 be high when H2 is also high, and vice versa. Using this feature, the Double Range Coder can be applied to H2 only, while H3 and H4 are dealt with in a different way, as seen in the following Sections.

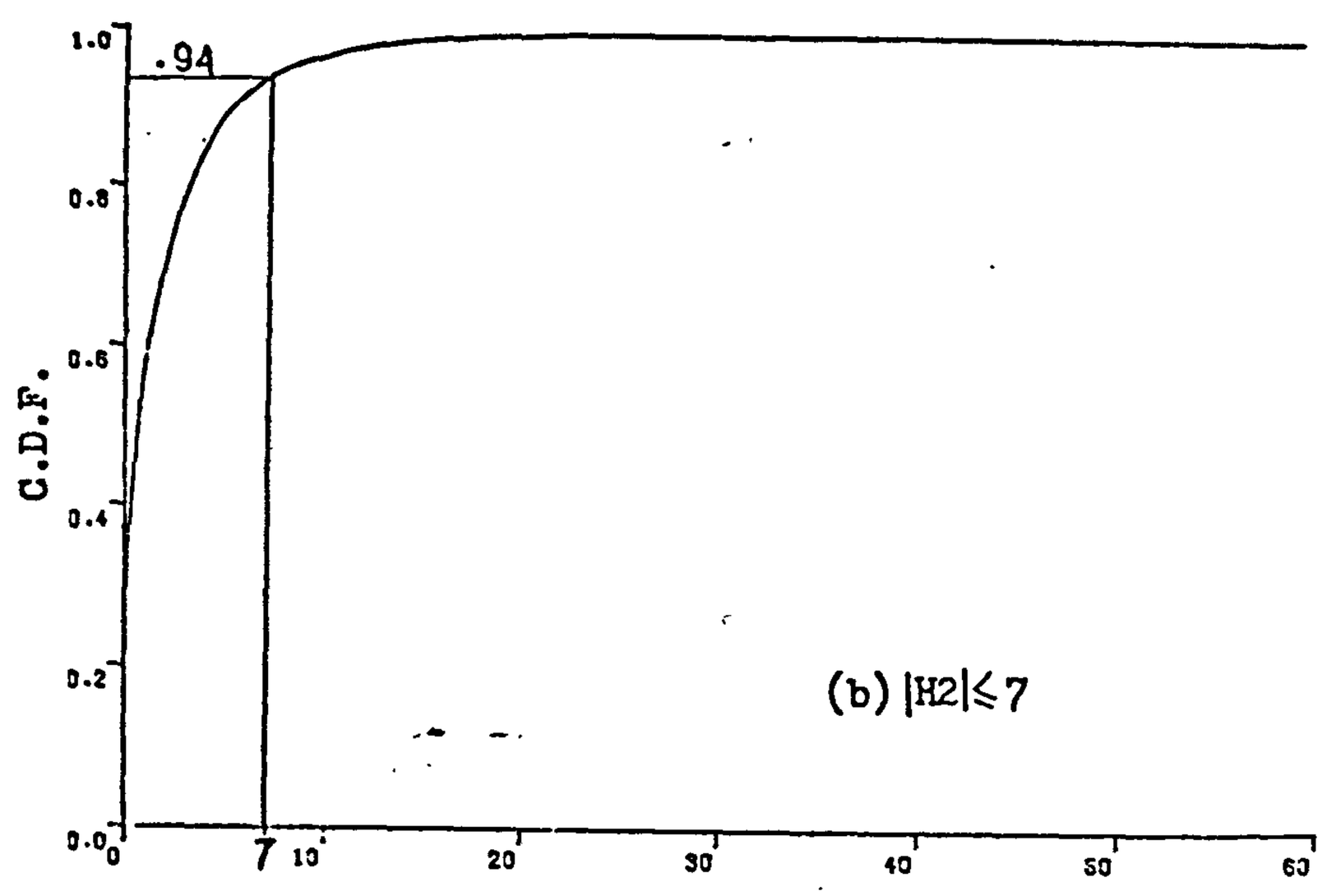
7.2. Conditional Statistics of Low Sequency Subvector :

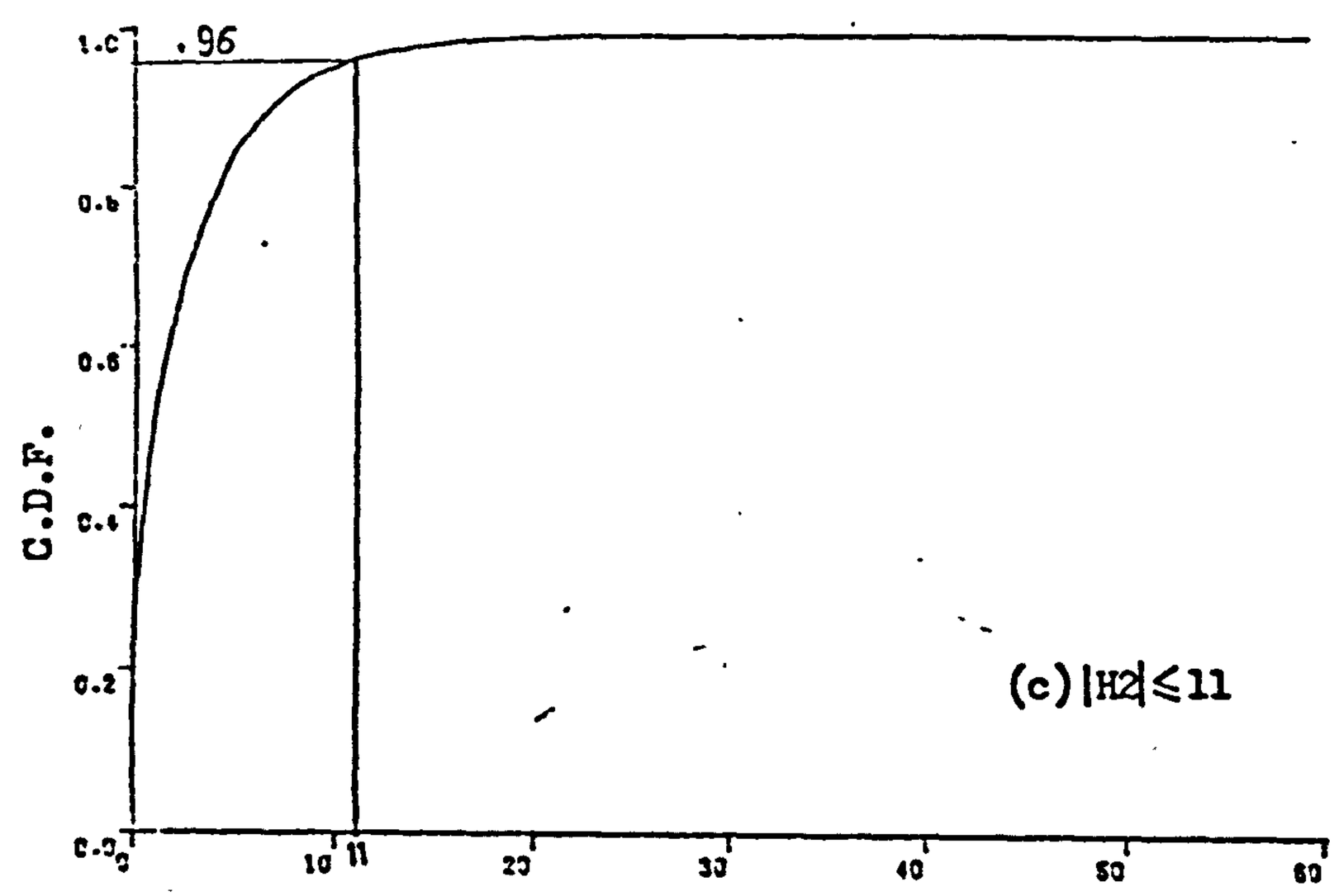
Figure (7.1) shows the conditional probability cumulative functions of absolute value of either one of coefficients H3 or H4, given that |H2| is equal to or less than a threshold value. Conditional characteristics for the sum of absolute values |H3| + |H4| are also shown in Figure (7.2). Four representative values of |H2| threshold were selected, namely 4, 7, 11, and 15. The graphs show that a degree of correlation still exists in the transform domain, and that the coefficients are not entirely independent in their values from each other. It is clear from figure , that in the majority of cases, values of both |H3| and |H4| , or even the value of their sum |H3| + |H4| , do not exceed the value of |H2| itself. Therefore, different coders for H3 and H4 will be used with |H2| high and |H2| low as explained now.

Whenever the value of |H2| is equal to or less than the threshold, values of |H3| and |H4| will normally be low as well, and then while coder " A " is used to code H2, a so called 'Compressed range coder' will be used for H3 and H4. This coder will have smaller quantizing steps as the range of coefficient values is itself low. If the value of |H2| exceeds the threshold value, both H3 and H4, (or one of them), may also have large values, then coder B is used to code H2 while an 'Expanded range coder' will be used to code H3 and H4. This allows increasing accuracy in coding the whole of the low sequency subvector coefficients, while increasing the total number of coders only by a factor of two instead of eight.

Both compressed and expanded range coders are based on conditinal statistical characteristics of both H3 and H4, given that |H2| is equal to or less than the preset threshold.







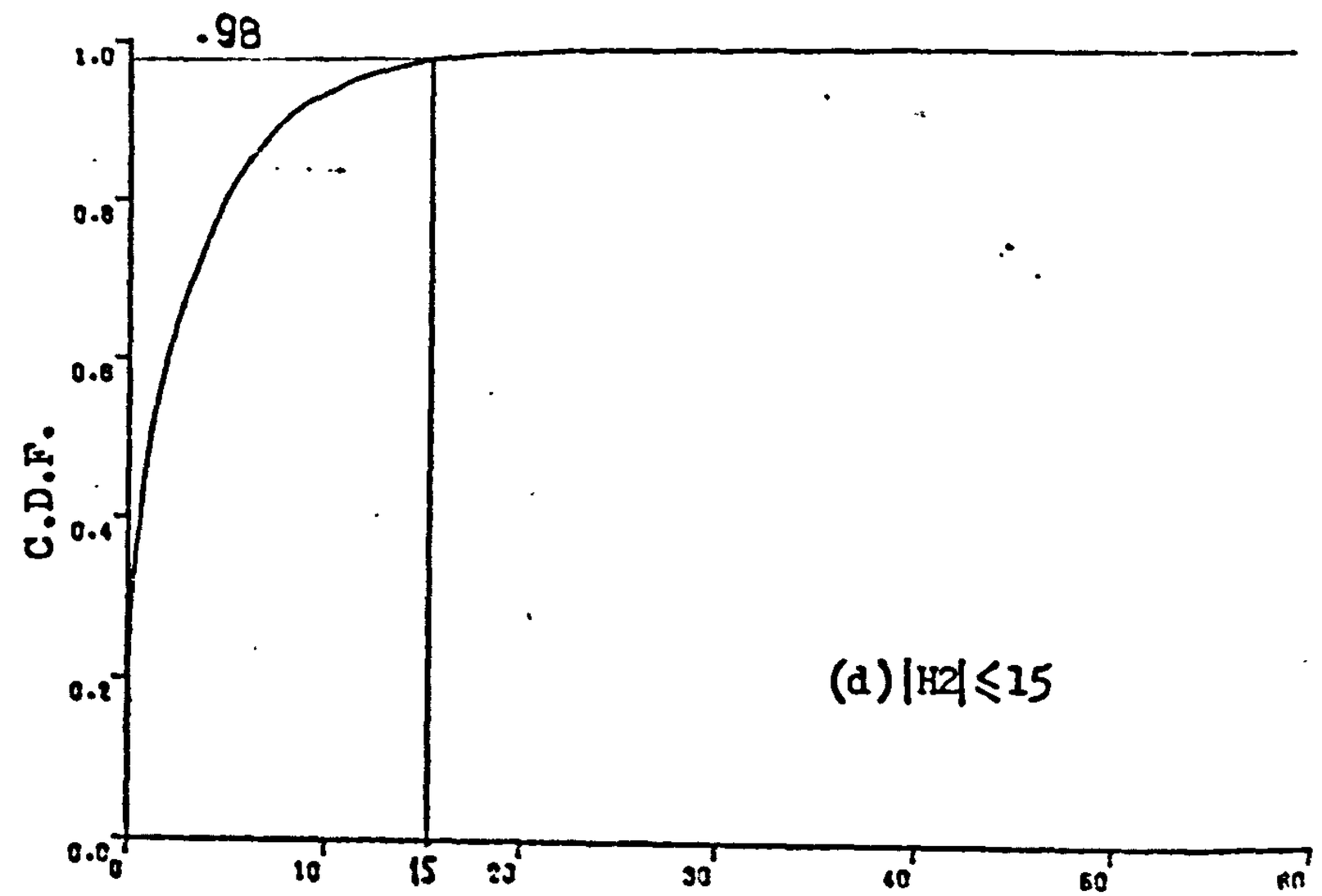
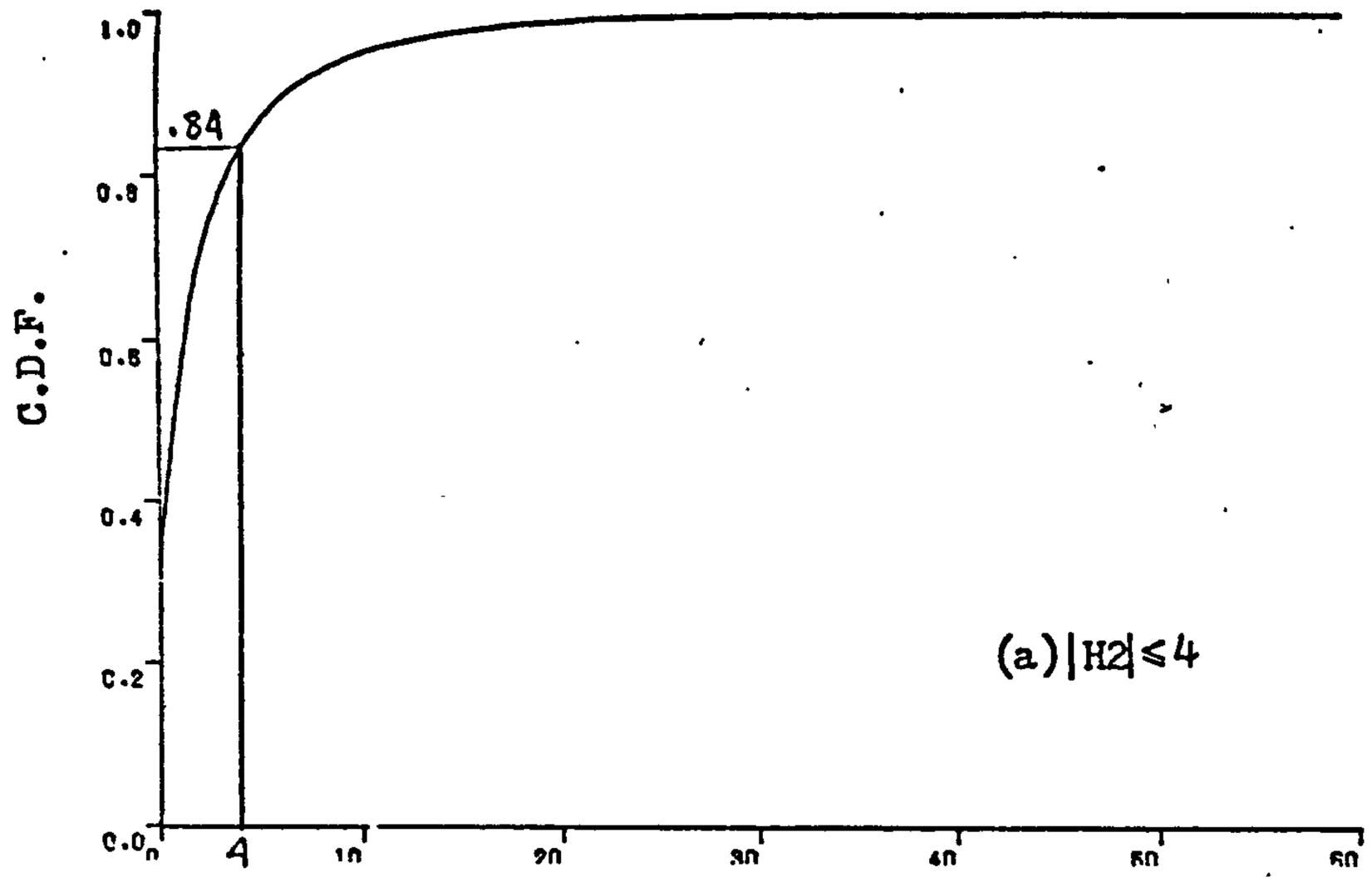
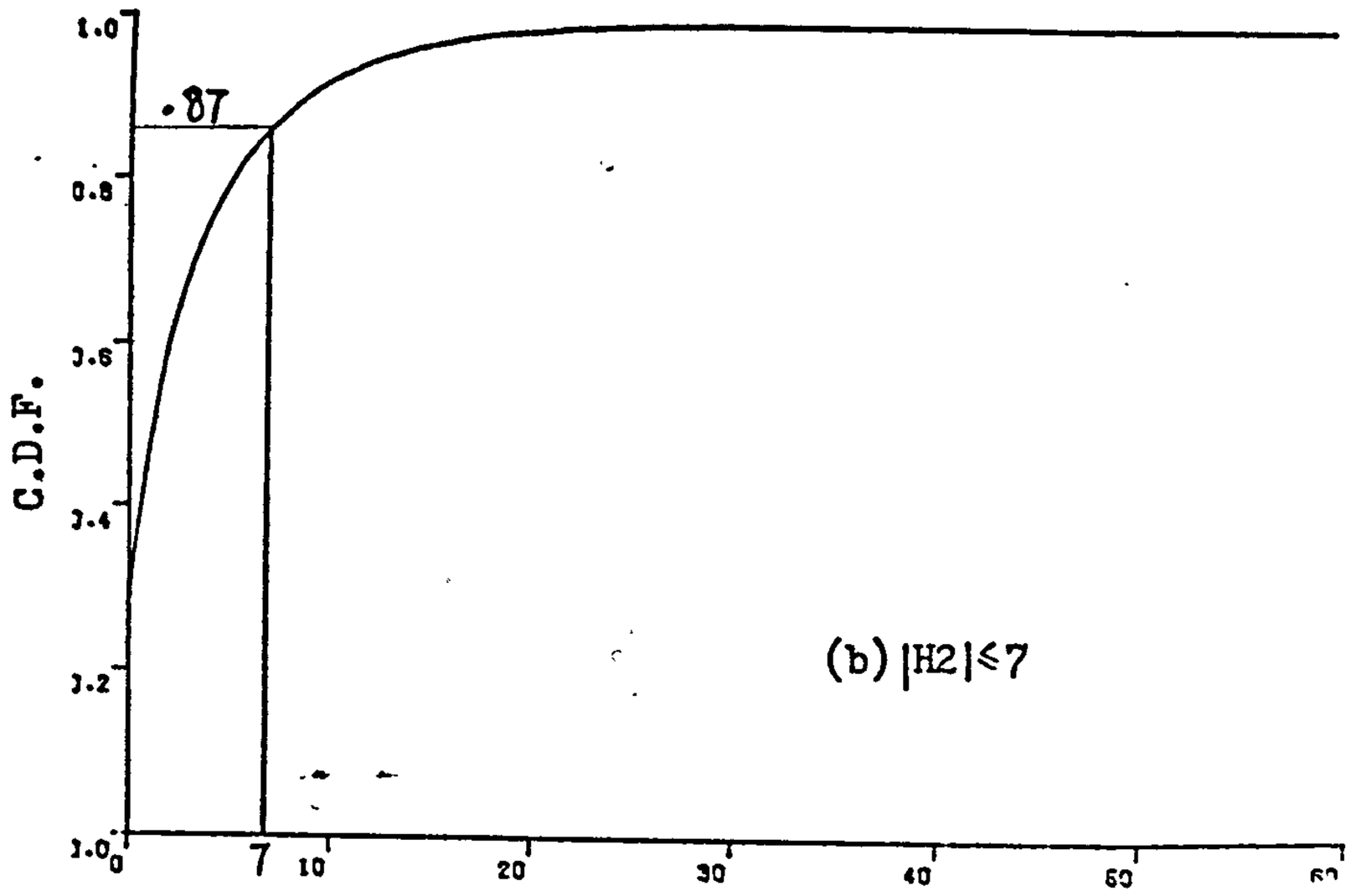
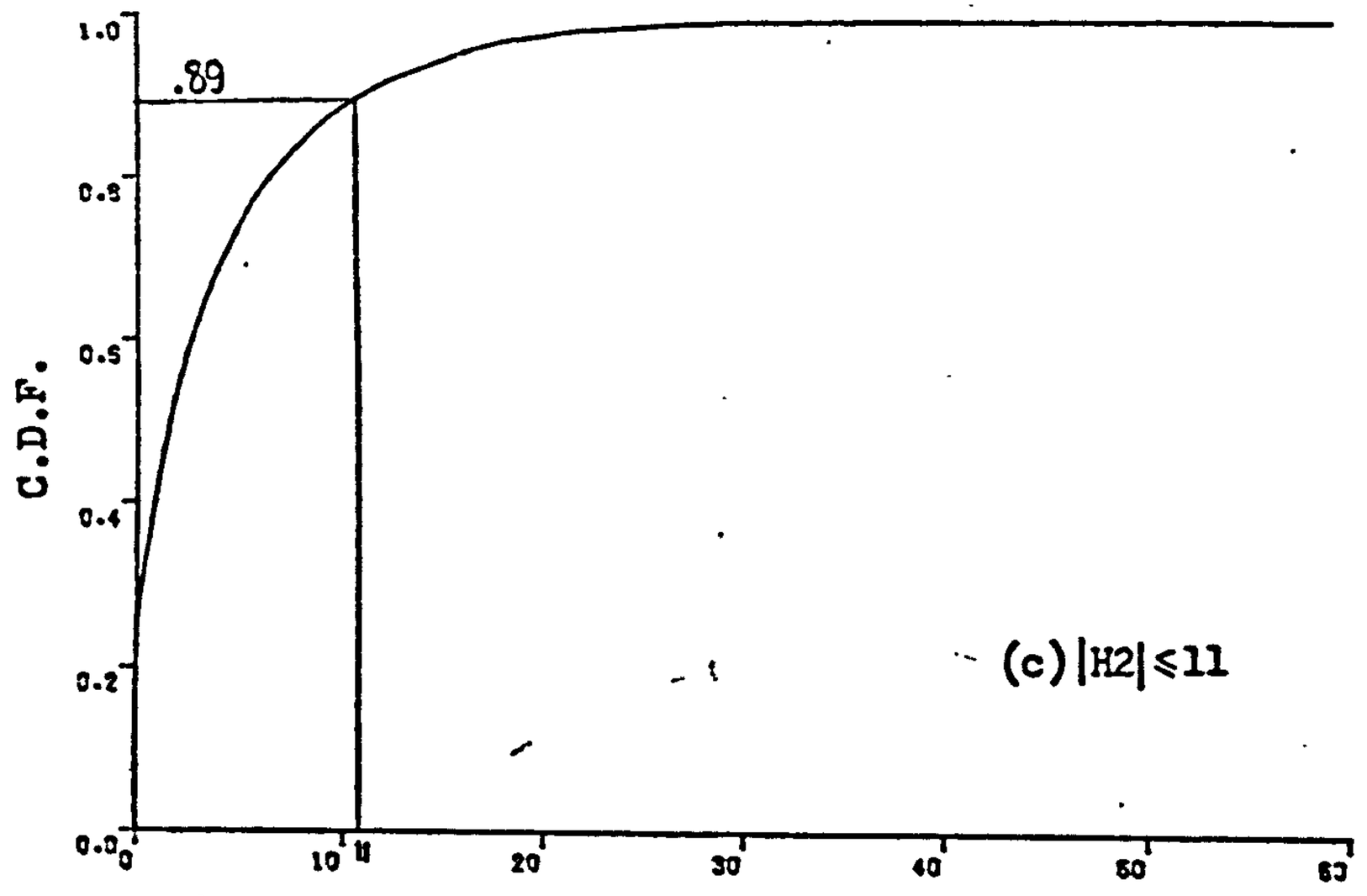


Figure (7.1). Conditional characteristics of either |H3| or |H4|, given |H2| ≤ a threshold value.







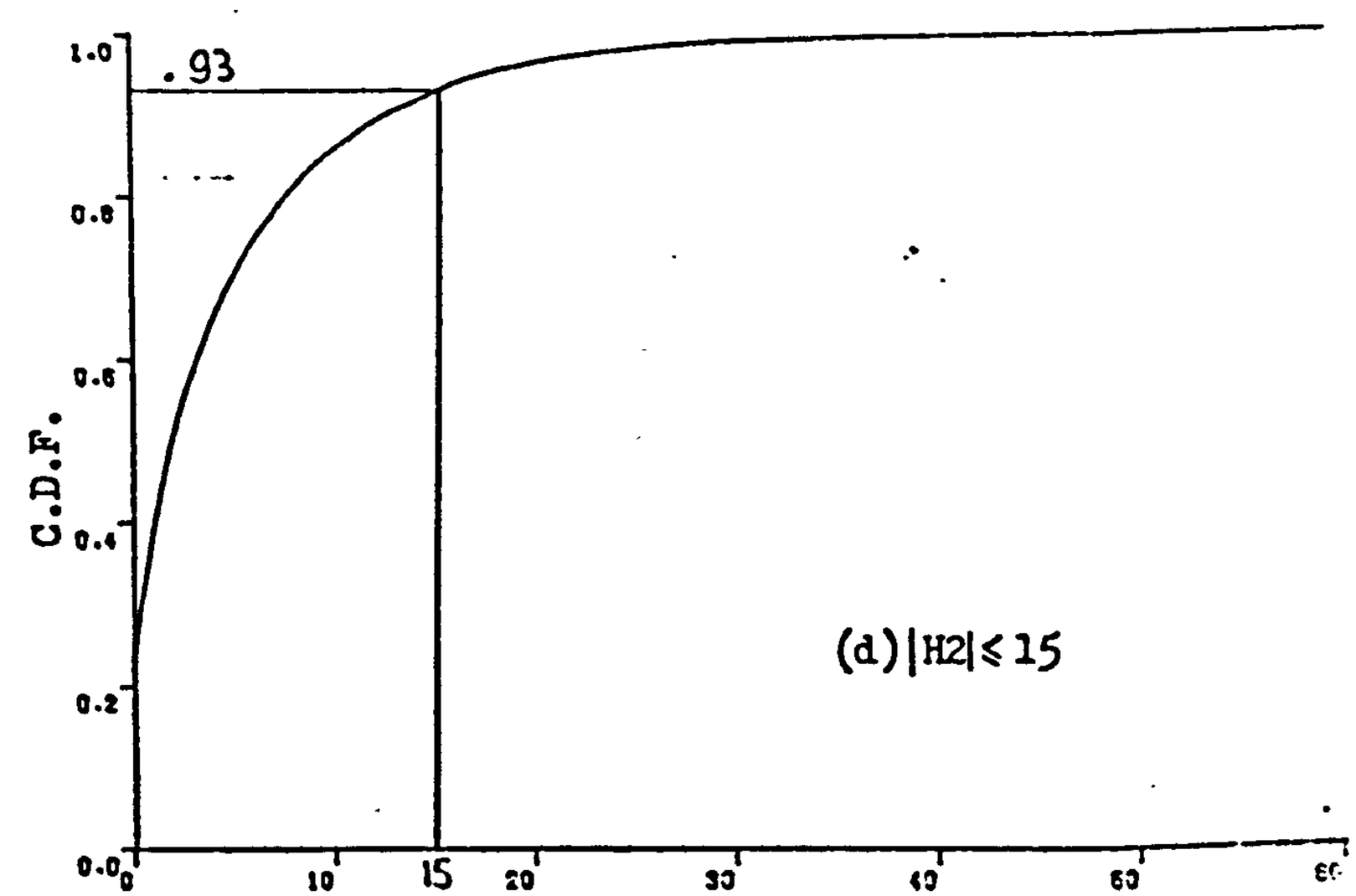


Figure (7.2). Conditional characteristics of $|H3|+|H4|$, given $|H2| \leq$ a threshold value.

7.3. Conditional Statistics of Medium and High Sequency Subvectors :

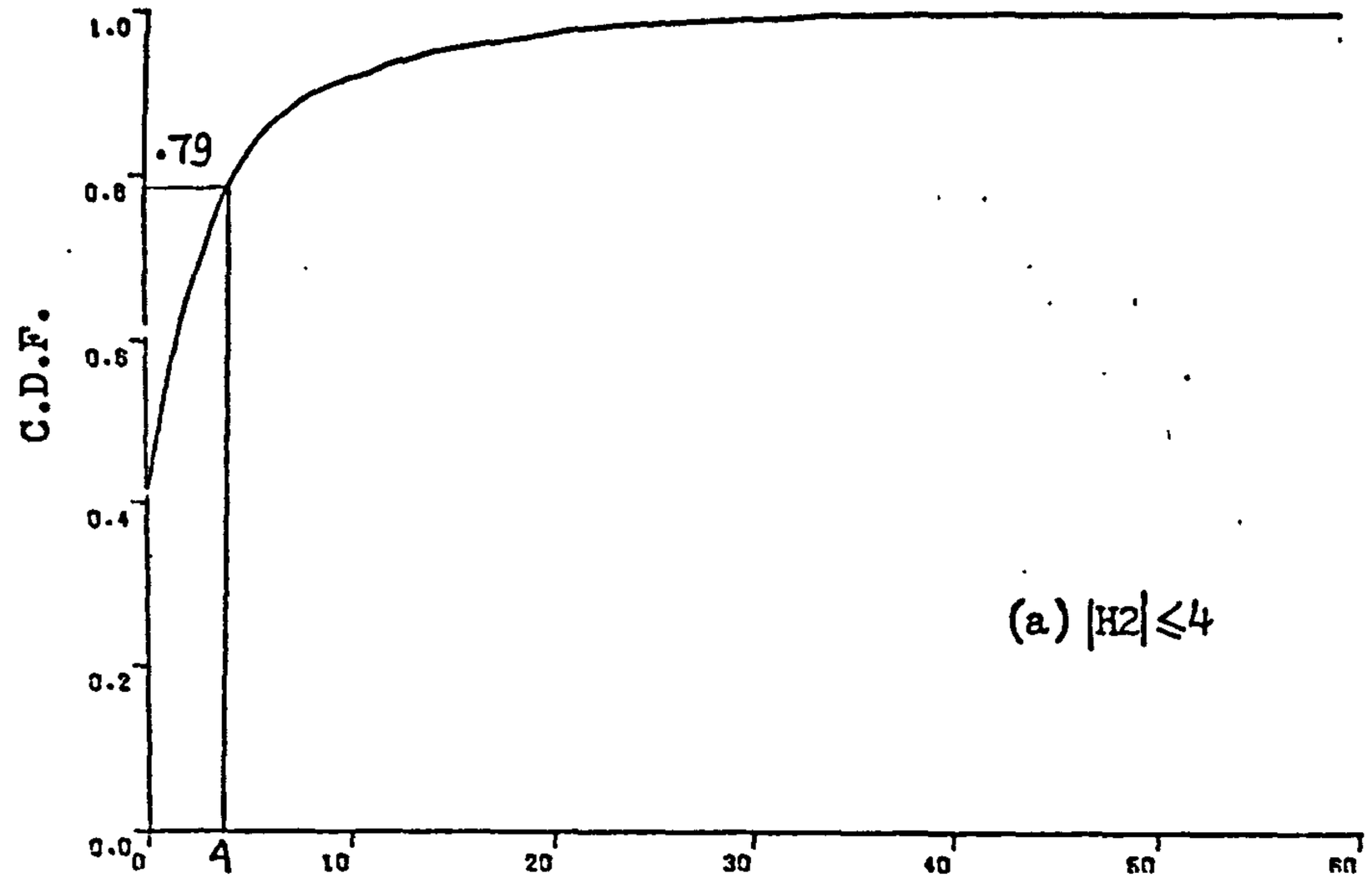
The conditional statistical characteristics for the remaining two subvectors, given a threshold value of |H2| ,can also be found. However, as the number of coefficients will be relatively large, graphs in Figures 7.3&7.4 show the characteristics for each subvector as a whole, for the same selected values of threshold as in Figures (7.1 and 7.2). In these graphs, the sum of the absolute values of all the coefficients in a subvector is considered.

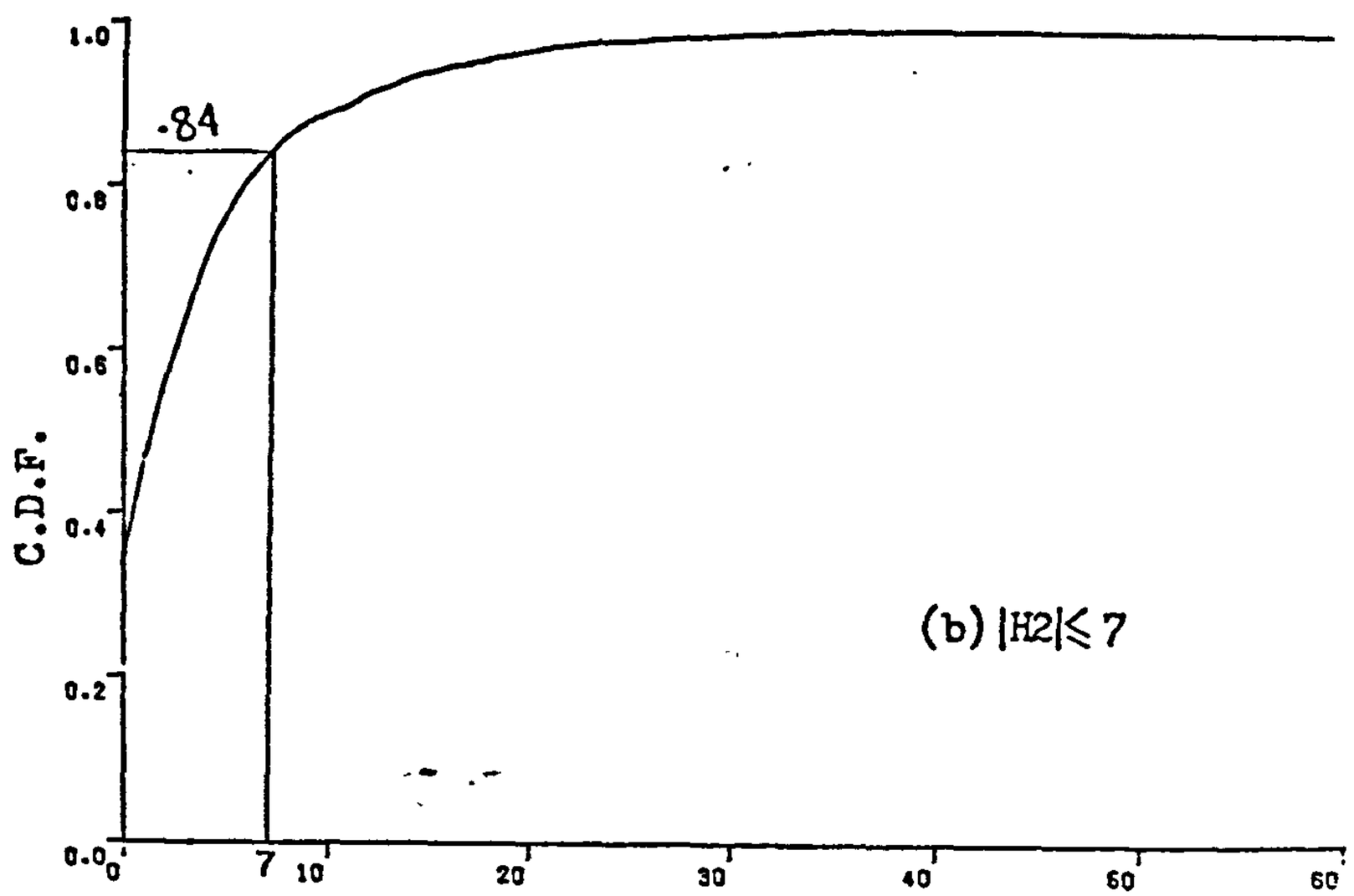
Like in the case of low sequency subvector, the graphs show that in the majority of cases, the total sum of coefficients in a subvector does not exceed the value of |H2| .Obviously, this proportion of cases increases as the the value of the threshold increases. It is also clear that the proportion of cases where the absolute value of any particular coefficilent exceeds the value of |H2| will be greatly reduced.

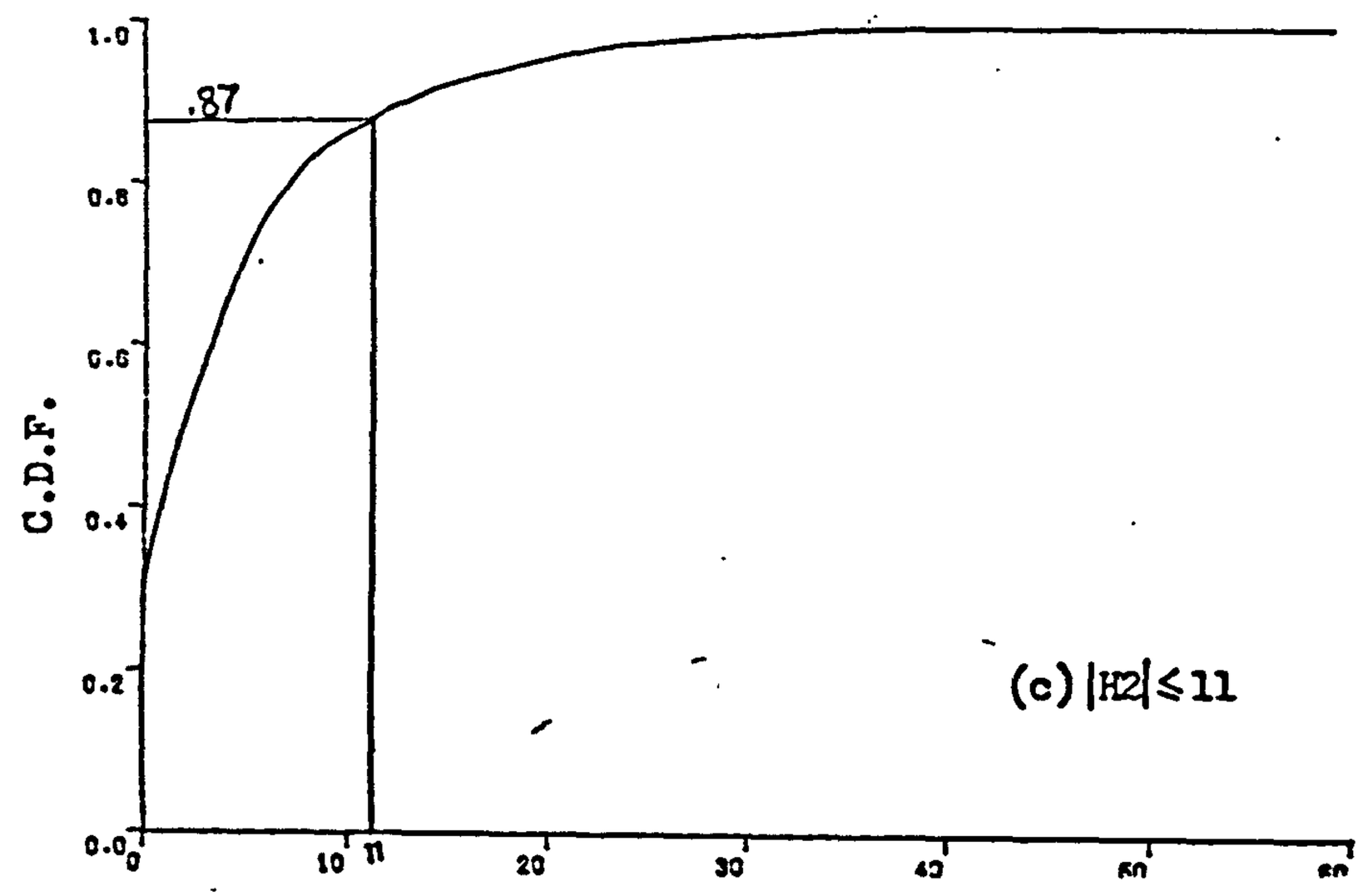
Tables (7.1. - 7.4.) show a detailed set of characteristics of all ac coefficients for some selected values of |H2| . The tables show the averages of absolute values of coefficients and their standard deviations in both cases of |H2| low and |H2| high.

From figures and tables, very wide variations in the coefficients' characteristics are observed between |H2| low and |H2| high. As averages and standard deviations of all coefficients vary with |H2| being low or high, different coders may be used for each coefficient (or a group of coefficients) for each state of |H2|. These coders will be called, as before, 'Compressed range coders' in case of H2 low, and 'Expanded range coders' in case of H2 high.

Guided by conditional characteristics from the above mentioned tables, a complete set of coders which can be used in conjunction with Directing Index D1,for example, is shown in Tables (7.5 - 7.20).In this set, the doubled number of coders will be 16.







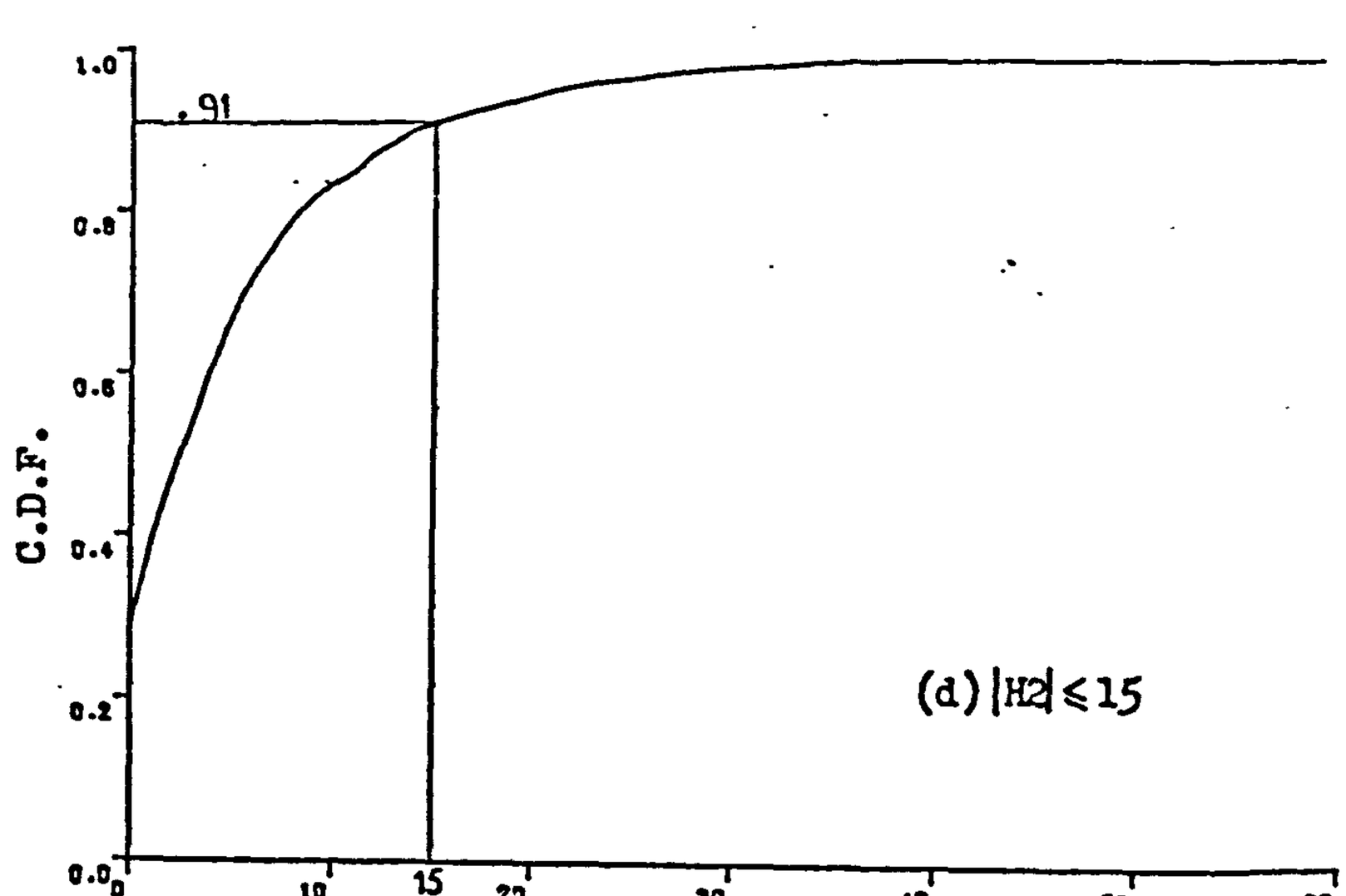
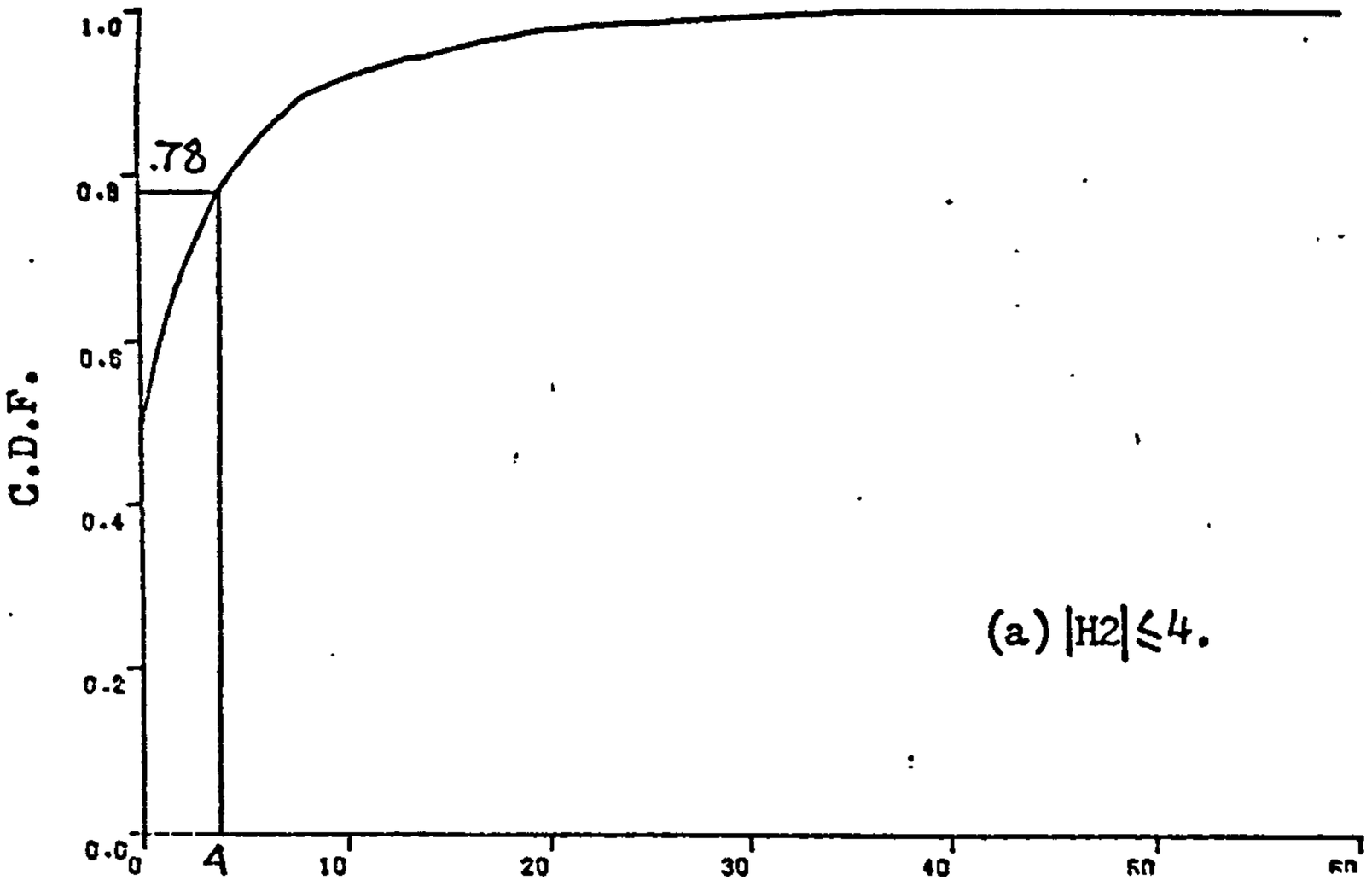
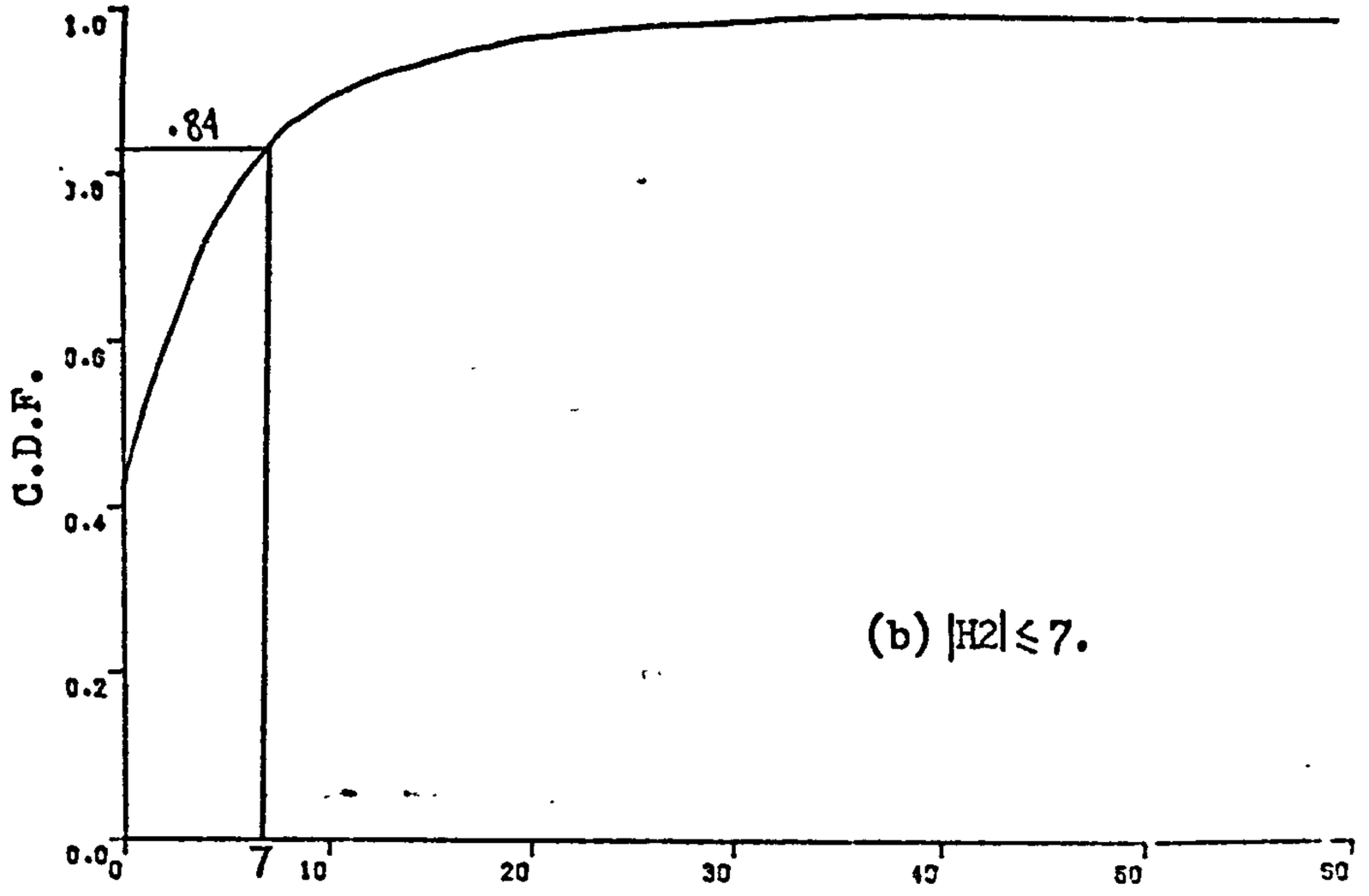


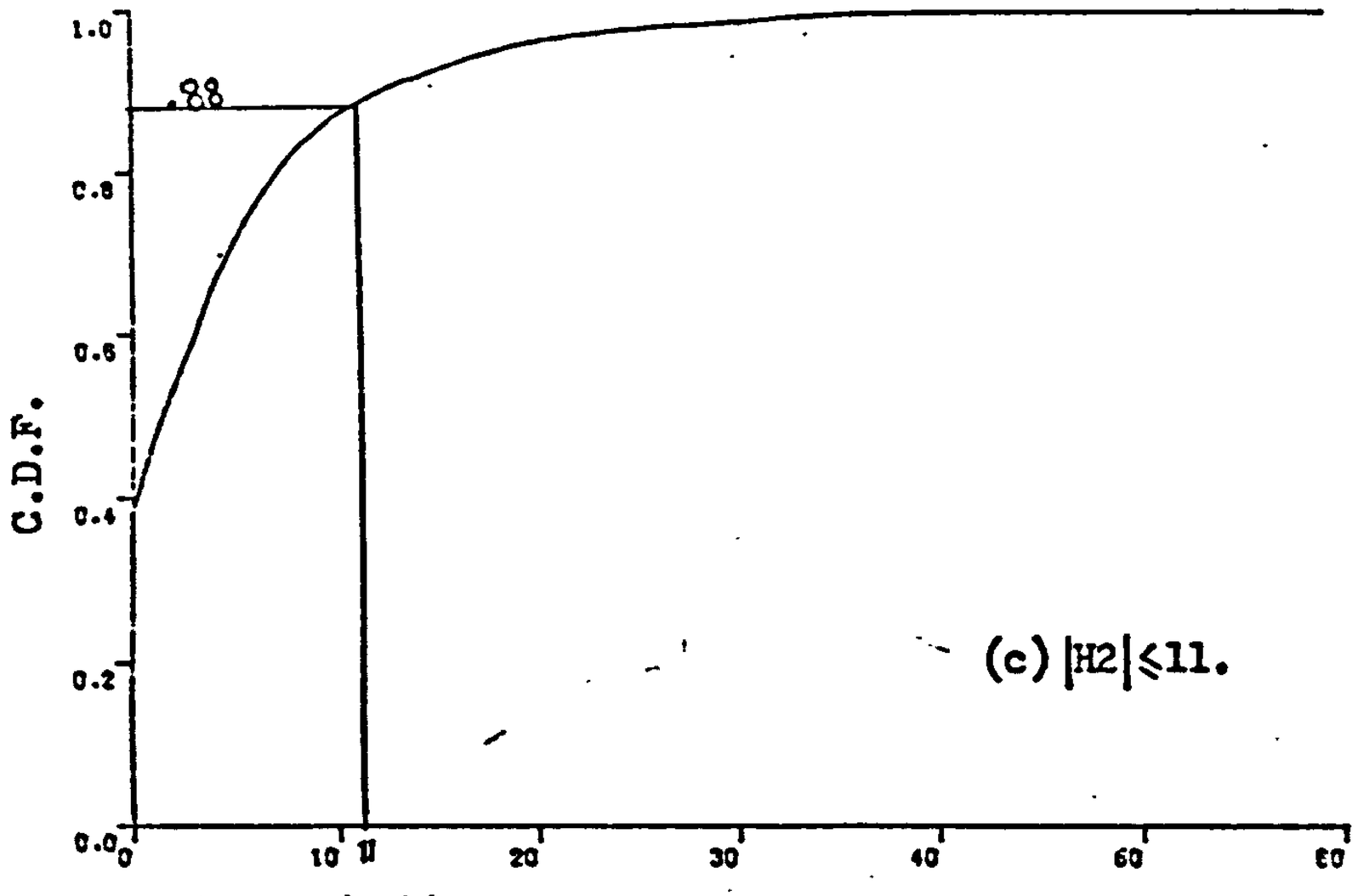
Figure (7.3). Conditional characteristics of the sum of medium sequency |coefficients|, given |H2|≤a threshold value.



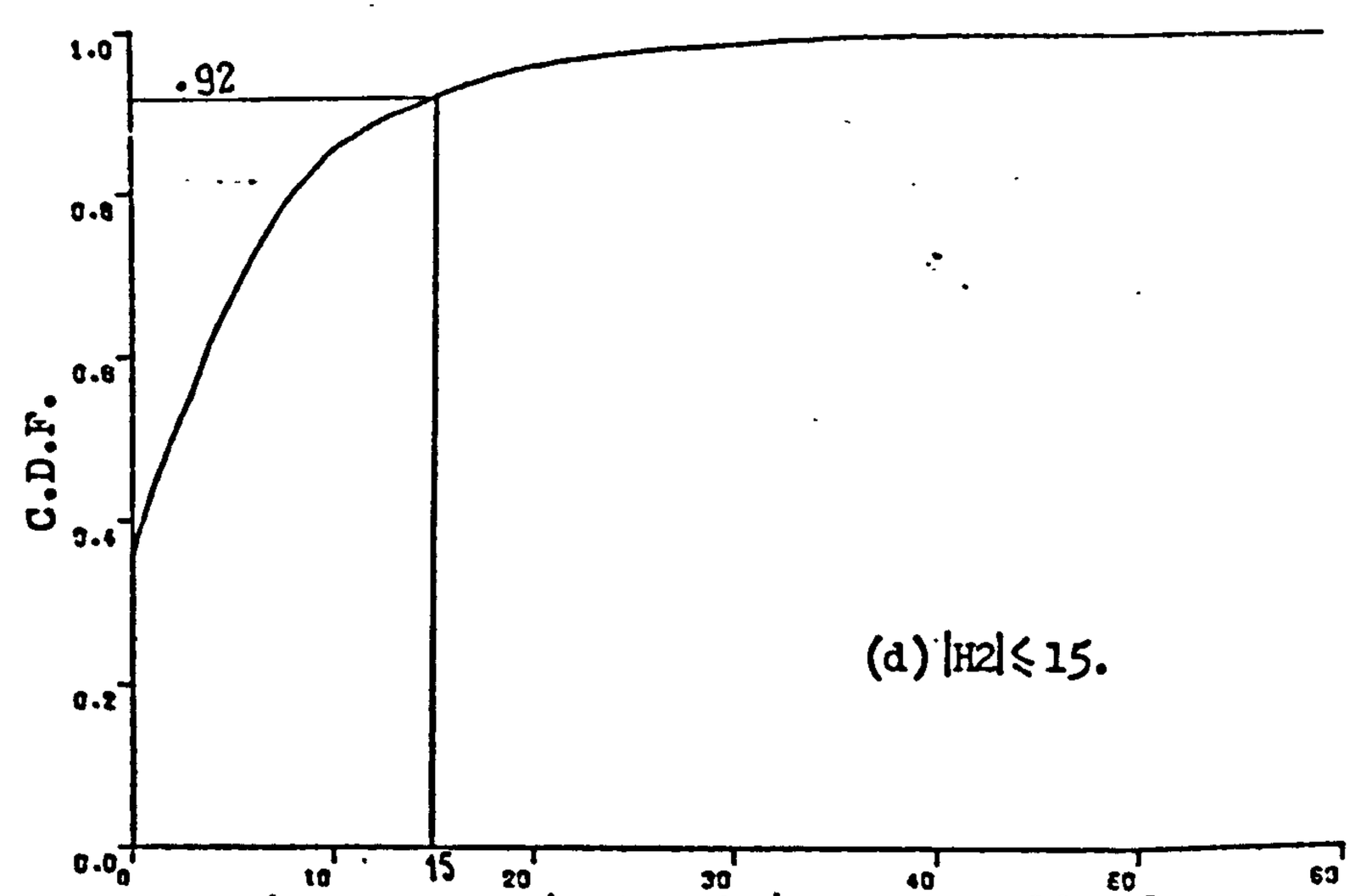
(a) $|H2| \leq 4$.



(b) $|H2| \leq 7$.



(c) $|H2| \leq 11$.



(d) $|H2| \leq 15$.

Figure (7.4). Conditional characteristics of the sum of high sequency |coefficients|, given $|H2| \leq$ a threshold value.

Table (7.1). Conditional statistical characteristics of coefficients, given |H2| does not exceed 4.

Threshold = 4.	|H2| low: 76.84 %.	|H2| high: 23.16 %.		
Coeff.	Average of absolute value.		Standard deviation.	
	|H2| Low.	|H2| High.	|H2| Low.	|H2| High.
2	1.4	15.0	1.7	17.7 *
3	2.0	8.6	2.7	12.1
4	1.9	8.5	2.4	11.0
5	1.3	3.6	2.4	5.6
6	1.1	4.3	1.6	6.3
7	1.1	3.8	1.7	5.7
8	1.0	3.9	1.6	5.2
9	0.6	1.3	1.4	2.6
10	0.6	1.4	1.0	2.7
11	0.6	1.6	1.1	2.7
12	0.6	1.6	1.1	2.6
13	0.6	1.7	1.1	2.7
14	0.5	2.0	0.8	2.9
15	0.5	1.8	0.7	2.6
16	0.4	1.8	0.5	2.4

* Note: standard deviation in this case is that of a new variable h2 = (H2 ∓ Threshold), as this is the difference which will be added to the constant, uncoded threshold. The minus sign is for positive H2 and the plus sign is for negative H2.

Table (7.2). Conditional statistical characteristics of ac coefficients, given |H2| does not exceed 7.

Threshold = 7.		|H2| low: 86.42 %.	|H2| high: 13.58 %.	
Coeff.	Average of absolute value.		Standard deviation.	
	|H2| Low.	|H2| High.	|H2| Low.	|H2| High.
2	2.2	17.4	2.6	19.8 *
3	2.5	10.0	3.3	14.9
4	2.3	10.1	2.8	13.7
5	1.5	3.9	2.6	6.5
6	1.3	4.9	1.9	7.7
7	1.3	4.4	2.0	6.8
8	1.2	4.7	1.8	6.3
9	0.64	1.4	1.6	2.7
10	0.62	1.7	1.1	3.0
11	0.63	1.8	1.2	3.0
12	0.68	1.7	1.2	2.9
13	0.70	1.8	1.2	3.1
14	0.61	2.3	0.91	3.5
15	0.59	2.1	0.87	3.2
16	0.47	2.3	0.63	3.0

* See note on Table (7.1).

Table (7.3). Conditional statitical characteristics of ac coefficients, given |H2| does not exceed 11.

Threshold = 11.		|H2| low: 94.04 %.	|H2| high: 5.96 %.	
Coeff.	Average of absolute value.		Standard deviation.	
	|H2| Low.	|H2| Hogh.	|H2| Low.	|H2| High.
2	3.1	21.12	3.8	24.1 *
3	3.04	10.8	3.8	20.8
4	2.92	10.9	3.25	19.4
5	1.67	4.23	2.77	8.69
6	1.56	6.12	2.16	10.53
7	1.54	4.98	2.22	9.15
8	1.43	5.32	1.93	8.88
9	0.69	1.60	1.60	3.47
10	0.67	2.06	1.28	3.76
11	0.71	2.09	1.30	4.03
12	0.75	2.02	1.30	3.75
13	0.77	2.07	1.32	4.01
14	0.71	3.00	1.02	4.89
15	0.69	2.36	0.97	4.34
16	0.60	2.58	0.72	4.20

* See note on Table (7.1).

Table (7.4). Conditional statistical characteristics of ac coefficients, given |H2| does not exceed 15.

Threshold = 15.	|H2| low: 97.11 %.	|H2| high: 2.89 %.

Coeff.	Average of absolute value.		Standard deviation.	
	|H2| Low.	|H2| High.	|H2| Low.	|H2| High.
2	3.72	23.69	4.94	27.46 *
3	3.31	9.78	4.34	26.84
4.	3.20	10.08	3.75	25.41
5	1.74	4.62	2.93	10.94
6	1.67	7.45	2.40	13.72
7	1.64	5.38	2.42	11.42
8	1.53	5.91	2.11	11.32
9	0.72	1.66	1.61	4.48
10	0.71	2.29	1.32	4.93
11	0.76	1.87	1.34	5.28
12	0.79	1.89	1.35	4.62
13	0.81	2.16	1.39	5.09
14	0.76	3.57	1.13	6.39
15	0.74	2.54	1.07	5.44
16	0.65	2.83	0.83	5.48

* See note on Table (7.1).

7.4. Directing Index D1 and Full Coders :

Procedures of computing Directing Index D1 in this case are almost the same as in Chapter 5. The only difference is the inclusion of another threshold and a comparison step for the value of H2. If |H2| is equal to or less than a preset threshold, an additionally assigned weighting factor will be 0, while if|H2| exceeds that threshold, this weighting factor will be 8. Adding the new weighting factor to the elements constituting the directing index D1, the entire range of 1 - 16 could now be obtained, with Index values (9 - 16) representing the case where |H2| is high.

Details of coders are as shown in Tables (7.5 - 7.20).

Table (7.5) Coders to be used with Directing Index D1=1.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H4	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 256
	Representatives	± 1 5
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.6). Coders to be used with Directing Index D1-2.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 4 256
	Representatives	± 1 5
H4	Quant. levels	± 0 4 256
	Representatives	± 1 5
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	± 0 2 256
	Representatives	± 0 3
H13	Quant. levels	± 0 2 256
	Representatives	± 0 3
H14	Quant. levels	± 0 2 256
	Representatives	± 0 3
H15	Quant. levels	± 0 2 256
	Representatives	± 0 3
H16	Quant. levels	± 0 2 256
	Representatives	± 0 3

Table (7.7). Coders to be used with Directing Index D1-3.

H2	Quant. levels	± 0 1 3 5 8
	Representatives	± 0 1 3 6
H3	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H4	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H5	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H6	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H7	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H8	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.8). Coders to be used with Directing Index D1-4.

H2	Quant. levels	± 0 1 3 5 8
	Representatives	± 0 1 3 6
H3	Quant. levels	± 0 4 256
	Representatives	± 1 5
H4	Quant. levels	± 0 4 256
	Representatives	± 1 5
H5	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 256
	Representatives	± 1 5
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	± 0 2 256
	Representatives	± 0 3
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.9). Coders to be used with Directing Index D1= 5.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 1 2 4 6 8 11 14 256
	Representatives	± 0 1 2 4 6 9 12 15
H4	Quant. levels	± 0 1 2 4 6 8 11 14 256
	Representatives	± 0 1 2 4 6 9 12 15
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 256
	Representatives	± 1 5
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.10). Coders to be used with Directing Index D1= 6.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H4	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	± 0 2 256
	Representatives	± 0 3
H13	Quant. levels	± 0 2 256
	Representatives	± 0 3
H14	Quant. levels	± 0 2 256
	Representatives	± 0 3
H15	Quant. levels	± 0 2 256
	Representatives	± 0 3
H16	Quant. levels	-
	Representatives	-

Table (7.11). Coders to be used with Directing Index D1= 7.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 1 2 4 6 8 11 14 256
	Representatives	± 0 1 2 4 6 9 12 15
H4	Quant. levels	± 0 1 2 4 6 8 11 14 256
	Representatives	± 0 1 2 4 6 9 12 15
H5	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H6	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H7	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H8	Quant. levels	± 0 2 5 8 256
	Representatives	± 0 3 6 9
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.12). Coders to be used with Directing Index D1= 8.

H2	Quant. levels	± 0 1 2 3 4 5 6 7 8
	Representatives	± 0 1 2 3 4 5 6 7
H3	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H4	Quant. levels	± 0 3 6 9 256
	Representatives	± 1 4 7 11
H5	Quant. levels	± 0 3 256
	Representatives	± 1 5
H6	Quant. levels	± 0 3 256
	Representatives	± 1 5
H7	Quant. levels	± 0 3 256
	Representatives	± 1 5
H8	Quant. levels	± 0 3 256
	Representatives	± 1 5
H9	Quant. levels	± 0 2 256
	Representatives	± 0 3
H10	Quant. levels	± 0 2 256
	Representatives	± 0 3
H11	Quant. levels	± 0 2 256
	Representatives	± 0 3
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.13). Coders to be used with Directing Index D1= 9.

H2	Quant. levels	± 8 11 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H4	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H5	Quant. levels	± 0 6 256
	Representatives	± 2 9
H6	Quant. levels	± 0 6 256
	Representatives	± 2 9
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	± 0 6 256
	Representatives	± 2 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	± 0 4 256
	Representatives	± 1 5
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.14). Coders used with Directing Index D1= 10.

H2	Quant. levels	± 8 11 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 6 256
	Representatives	± 2 9
H4	Quant. levels	± 0 6 256
	Representatives	± 2 9
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	± 0 4 256
	Representatives	± 1 5
H12	Quant. levels	± 0 4 256
	Representatives	± 1 5
H13	Quant. levels	± 0 4 256
	Representatives	± 1 5
H14	Quant. levels	± 0 4 256
	Representatives	± 1 5
H15	Quant. levels	± 0 4 256
	Representatives	± 1 5
H16	Quant. levels	± 0 4 256
	Representatives	± 1 5

Table (7.15). Coders to be used with Directing Index D1-11

H2	Quant. levels	± 8 14 21 30 256
	Representatives	± 10 16 24 34
H3	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H4	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H5	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H6	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H7	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H8	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.16). Coders to be used with Directing Index DI= 12.

H2	Quant. levels	± 8 14 21 30 256
	Representatives	± 10 16 24 34
H3	Quant. levels	± 0 6 256
	Representatives	± 2 9
H4	Quant. levels	± 0 6 256
	Representatives	± 2 9
H5	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H6	Quant. levels	± 0 6 256
	Representatives	± 2 9
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	± 0 6 256
	Representatives	± 2 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	± 0 4 256
	Representatives	± 1 5
H12	Quant. levels	± 0 4 256
	Representatives	± 1 5
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.17). Coders to be used with Directing Index D1= 13.

H2	Quant. levels	± 8 11 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H4	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H5	Quant. levels	± 0 6 256
	Representatives	± 2 9
H6	Quant. levels	± 0 6 256
	Representatives	± 2 9
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	± 0 6 256
	Representatives	± 2 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.18). Coders to be used with Directing Index DI= 14.

H2	Quant. levels	± 8 1 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H4	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H5	Quant. levels	-
	Representatives	-
H6	Quant. levels	-
	Representatives	-
H7	Quant. levels	-
	Representatives	-
H8	Quant. levels	-
	Representatives	-
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	± 0 4 256
	Representatives	± 1 5
H12	Quant. levels	± 0 4 256
	Representatives	± 1 5
H13	Quant. levels	± 0 4 256
	Representatives	± 1 5
H14	Quant. levels	± 0 4 256
	Representatives	± 1 5
H15	Quant. levels	± 0 4 256
	Representatives	± 1 5
H16	Quant. levels	-
	Representatives	-

Table (7.19). Coders to be used with Directing Index D1= 15.

H2	Quant. levels	± 8 11 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H4	Quant. levels	± 0 3 6 10 15 20 26 33 256
	Representatives	± 1 4 7 12 17 22 28 35
H5	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H6	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H7	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H8	Quant. levels	± 0 4 9 15 256
	Representatives	± 1 6 11 17
H9	Quant. levels	-
	Representatives	-
H10	Quant. levels	-
	Representatives	-
H11	Quant. levels	-
	Representatives	-
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

Table (7.20). Coders to be used with Directing Index DI= 16.

H2	Quant. levels	± 8 11 14 18 23 28 34 41 256
	Representatives	± 9 12 15 20 25 30 36 44
H3	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H4	Quant. levels	± 0 6 13 22 256
	Representatives	± 2 8 16 26
H5	Quant. levels	± 0 6 256
	Representatives	± 2 9
H6	Quant. levels	± 0 6 256
	Representatives	± 2 9
H7	Quant. levels	± 0 6 256
	Representatives	± 2 9
H8	Quant. levels	± 0 6 256
	Representatives	± 2 9
H9	Quant. levels	± 0 4 256
	Representatives	± 1 5
H10	Quant. levels	± 0 4 256
	Representatives	± 1 5
H11	Quant. levels	± 0 4 256
	Representatives	± 1 5
H12	Quant. levels	-
	Representatives	-
H13	Quant. levels	-
	Representatives	-
H14	Quant. levels	-
	Representatives	-
H15	Quant. levels	-
	Representatives	-
H16	Quant. levels	-
	Representatives	-

7.5. Results and Assessments :

Applying the coders of Tables (7.5 - 7.20) in conjunction with Directing Index D1 (with values from 1 to 16 in this case), has shown an improvement of about 1.5 dB in Signal-to-Noise Ratio over the results obtained from the ordinary single range coders of Tables (5.4 - 5.11), in Chapter 5. Although this improvement seems relatively low, and may not thought to justify the increased number of coders used, it should not discredit the potentiality of the scheme in improving the performance. More work involving coders optimisation, by changing different coding parameters, may well yield much better results. Such a work could be best done using a frame-store in addition to the transformer. This will speed up the operation and allow quick subjective assessments for different parameters. Another area of potential improvement is the possible application of two coders to more high dynamic range (low sequency) terms. For example, if an increase in the total number of coders of a factor 4 is allowed, then both H3 and H4 could be coded more finely as H2, with another increase in performance fidelity.

CHAPTER EIGHT

CONCLUSIONS

8.1. Review.

The work reported here is an investigation of Hadamard transform adaptive coding of digital television signals. The transform is one of several means of bit rate reduction in digital video transmission. Investigations of the transform aimed to exploit its characteristics in reducing the number of transmitted coefficients as well as determining the best ways of coding them. For efficient coding and bit rate reduction, the coding should be adaptive. For efficient adaptivity, coding schemes should be based on more detailed characteristics of transform domain obtained as a result of these investigations.

In the first part of this study, extensive investigations of transform domain coefficient characteristics are presented, with particular attention to possible contributions of some of these features to the ultimate target of bit rate reduction. In the second part, the results of the first are used to devise objective computable Directing Indexes to select the appropriate coder for each particular subpicture. These indexes have the objective of being related to the local characteristics of the subpictures. Not only the over all activity or busyness of the subpicture is considered, but also the level of activities of different divisions thereof. This avoids the major disadvantage of the present 'activity index', namely, the misclassification of some subpictures based on their over all activities.

As a general conclusion of the first part, concerning the transform domain investigations, the following features were apparent:

I. Monochrome Signals:

1. Line (one-dimensional) transform, generally, compacts more energy in fewer transform coefficients than block (two-dimensional) transform. In addition to this fact, the realisation of line transform is much easier and needs far less memory as there are no line storages required, as in case of two-dimensional, to build up the blocks.

2. Abruptancy in statistical characteristics, present in line transform domain, as seen in Chapter 2, makes it easier and virtually systematic to deal with coefficients belonging to the same division (subvector or subset).

3. In case of two-dimensional transform, there are no practical differences regarding the energy compaction, among cases of different block sizes or different block shapes. Moreover, for the same block sizes, reversed dimensions did not show any noticeable differences. This may well lead to a conclusion that, in any case of two-dimensional transform, blocks with smaller number of television lines (vertical dimension) should be used in order to save vast line storages requirements and block building complexities.

II. Colour Signals:

Due to the presence of subcarrier frequency, the energy of an image is likely to be spread over more transform coefficients than in the case of monochrome signals, which means less efficient compaction. However, careful conditioning of the spatial domain samples, and an efficient exploitation of their transform domain characteristics, could yield good results comparable with monochrome transform. This is done by rearranging the samples, prior to transformation in a way such that the redundancy is maximum, either by component isolation (in the case of components transforms), or by interlacing the samples one subcarrier interval apart (in case of the so called laced transform). This seemed to approximate the characteristics of the monochrome transform.

Investigation of directly transforming the composite signal (direct transformation), has shown that while the most inefficient process was that of direct transform at thrice the subcarrier frequency sampled signals, due to the spread of image energy among more different transform coefficients, the same type of transformation at a sampling

frequency of four times the subcarrier seemed extremely efficient. The excellent energy compaction property of direct transformation at $4f_{sc}$, due to similarities and simplicities of sample values and its full suitability to the differencing processes involved in Hadamard transform, has made it more promising. From the results of Chapter 3, it is even potentially possible to code the two mid-sequency terms, in addition to the dc term, to get a reasonably recovered image, with substantial bit rate saving.

The problem of effects of limited hardware and coding wordlengths on the recovered image is also investigated. Although it may seem logical that as the transform size increases, more energy proportion will be compacted at lower sequency coefficients, hence allowing more bit rate reduction, it has been found that the inevitable normalizing process presents an error (noise) of its own,which increases with the transform size. In addition to this normalizing error, another source of noise will be the decrease in redundancy among spatial domain samples in an increased subpicture size. As a result of investigations of Chapter 4, an upper limit on the transform size was recommended to be in the range of (32 - 64) samples for a monochrome image.

In the second part of this study, directing indexes to be used for adaptive coding were devised, based on results of investigations of transform domain characteristics. Starting with the monochrome signals, all the three indexes suggested in Chapter 5 are based on interrelationships among different coefficients and groups of coefficients in the transform domain. The first index, D1, being the simplest of all, allows simultaneous assessment of activity of each subvector in the transform, and hence direct the subpicture transform to its most appropriate coder (and decoder). Although a total number of 8 coders (16 in another stage) were used here, this number could be increased as far as the realisation

process can reach. This will involve further dividing of the activity levels of different subvectors, and hence better adaptivity in coding schemes and parameters. Tests show an increase in SNR of about 6 dB over non-adaptive coding using a single general coder based on statistical characteristics. Due to the lack of published results concerning any tests using the existing activity index, no specific comparison was possible. However, with an experiment based on an index similar to,that index, and using the Discrete Cosine Transform rather than Hadamard transform, the comparison shown an improvement in performance at a comparable bit rate. Another assessment using a purely hypothetical ideal directing process shows that a slight decrease of only about 1 dB is encounered.

The second index D2, involved the study of energy spectrum in transform domain. Distribution of energy among particular subsets in both medium and high sequency subvectors was studied. Only one subset in each subvector was coded, in some cases, to allow a fair consideration of all the constituent groups within the available limits of coding bits.

In the third index, D3, a deeper look to the problem of high sequency subvector is paid. This problem is that, the number of coefficients in this subvector is half the total number,and, consequently, the different possibilities of energy distribution among them are too much. Here, the subvector is further divided into four subsets in an attempt to isolate, more precisely, the area with maximum energy content.

Although the improvements in SNR's by using D2 or D3 seem very modest, at about 1 dB over D1, and hence may not justify the complexities of computations over D1, further investigations, involving optimization of coders used with each index, may yield more improvements.

As for the colour signals, both the laced and component transforms

were considered to be nearly similar to the monochrome transform, and therefore the same directing indexes are proposed to be applied, with some modifications involving the thresholds in computing the indexes. As an example, results of applying the three indexes to the luminance component transform were shown, in which satisfactory SNR's were achieved. However, due to the complexities involved with component transforms (being in fact three independent transforms), in addition to the larger number of coefficients (basically three times an ordinary transform terms), it is felt that its application may prove difficult in this area. Although the so called laced transform brings the selected samples of colour signals to the general form of a monochrome one, its need for a lacing process at the transmitting end and a delacing process at the receiving end may discredit its advantage.

In the case of the simple direct transformation, the promising characteristics of the $4f_{sc}$ sampling frequency mentioned before proved to be useful. As seen in Chapter 6, the direct transform at $4f_{sc}$ has shown about 12 dB improvement in SNR over the $3f_{sc}$ direct transform. With full compatibility regarding ' actual ' bit rate, the quality of the reconstructed image was still about 9 dB better than in case of $3f_{sc}$. The performance of the $3f_{sc}$ direct transform was, as expected, inferior to any other of the considered cases.

Directing indexes proposed and tested here have the major advantage of being independent from statistical characteristics of images, and hence can be used with any category of images. Although statistical studies were pwrformed to get their structure and to compute their values, in their applications they are entirely independent of these statistical features. No matter what proportion of total subpictures will be classified in a particular class, the tctal number of coding bits is constant for all cases. Arithmetic operations involved in computing these indexes are easy and straightforward, probably as the transform itself, and should not need excessive time delay.

The double range coder used to accurately code the high dynamic lowest sequency coefficient, H2, proved to have some potential advantage. More investigations of conditional statistics, and allowing higher number of coders may well yield much better results than the 1.5 dB increase achieved here for a single coefficient. With continuous advances in digital integrated circuit technology, the increase in the number of coding schemes should not be of any practical difficulty.

8.2. Suggestions for Future Work :

The following are some areas of possible further investigations.

1. Optimisation of the coders used here, in order to get the best possible results. A television frame store and an arithmetic processor, in addition to the Hadamard transformer, will facilitate fast and efficient changes of coding parameters, with instant subjective assessments.

2. Energy spectra of colour signal transforms could be studied in more detail in order to apply both the directing indexes D2 and D3. This will also involve determining the most appropriate values of thresholds.

3. More redundancy in spatial domain could be further exploited in one of three possible ways:

a). Increasing the vector size in a line transform, and applying the same principles considered here.

b). Investigations of two-dimensional transform, especially for small limited block sizes (as 4x4), although it may be expected, from results of Chapter 2, that no justifying improvement in performance would be obtainable.

c). The possibility of eliminating temporal redundancy among successive frames in an interframe system.[79] Such a system could lead to more bit rate reduction due to the nature of television images of relatively slow changes during a frame interval. This, of course, will involve vast storage requirements, more complex processing, and longer time delays.

4. As found in Chapter 3, the component transforms at $4f_{sc}$ have some sort of 'banding' in the energy packing characteristics. This could be of particular use in selecting some coefficients from each 'band' for coding. More investigations may be necessary to isolate the most active groups of coefficients, and also to see if this will balance the increased number of total coefficients in component transforms.

5. Directing indexes devised here were based on investigations of ac energy packing characteristics. Therefore, the entire scheme could still be used in a hybrid coding system, where ac coefficients are dealt with in the same way as explained here, while the dc term, H1, might be better coded in a DPCM.[79,80,82]

6. Extension of double range coders to more low sequency terms, based on more investigations of conditional characteristics and finding the appropriate values of thresholds for different coefficients.

REFERENCES

(1) Weaston,J.D., "Transformation of television by Pulse Code Modulation", Electrical Communications,vol. 42, Number 2,pp 165-173, 1967.

(2) Devereux, V.G., "Application of P.C.M. to broadcast quality video signals, Part 1, Subjective Study of the Coding Parameters", The Radio and Electronic Engineer, vol. 44, No.7,pp 373-381, July 1974.

(3) Moore,T.A., "Digital Video: number of bits per sample required for reference coding of luminance and colour-difference signals", BBC Research Report BBC RD 1974/42, Dec. 1974.

(4) Webster, D., "Digital picture coding", Wireless World, vol. 84, No. 1514, pp 67-70, Oct. 1978.

(5) Downing, O.J., and Gallagher, J.C., "A Feasibility Study of the Application of Digital Techniques to Television Studio Systems", Independent Television Companies Association Report 101/75, Sept.1975.

(6) Hotelling, H., "Analysis of complex statistical variables into principle components", J.Educ.Psychology, vol.24, pp 417-441, 498-520, 1933.

(7) Kramer,H.P.,and Mathews,M.V., "A linear coding for transmitting a set of correlated signals", IRE Trans. Inform. Theory, vol. IT-2, pp 41-46, Sept. 1956.

(8) Huang,J.J.Y.,and Schultheiss,P.M., "Block quantization of correlated Gaussian random variables", IEEE Trans. Commun. Syst., vol. CS-11, pp 289-296, Sept. 1963.

(9) Karhunen,H., "Uber lineare methoden in der Wahrscheinlich-Keitsrechnung", Ann.Acad.Sci.Fenn., Ser. A.I. 37, Helsinki, 1947. English translated in The RAND Corp., Doc. T-131, Aug.11, 1960, as "On linear Methods in probability theory",(trans. I.Selin).

(10) Ahmed, N., and Rao, K.R., 'Orthogonal Transforms for Digital Signal Processing', New York/ Heidelberg: Springer, 1975.

(11) Koschman,A., "On the filtring of nonstationary time series", in Proc. 1954 Nat.Electron.Conf., P. 126.

(12) Brown,J.L.Jr.,"Mean-square truncation error in series expansions of random functions", J.SIAM, vol. 8,pp. 18-32, Mar. 1960.

(13) Cooley, J.W.,and Tukey, J.W., "An algorithm for machine Calculation of Complex Fourier series", Mathematics of Computation, vol.19, No.90, pp. 297-301, 1956.

(14) Cochran, W.T.,et al., "What is the Fast Fourier Transform?", Proc. IEEE, vol. 55, No. 10, pp. 1664-1673, Oct.1967.

(15) Cooley, J.W., Lewis, P.A.W., and Welch, P.D., "Historical notes on the Fast Fourier Transform", Proc. IEEE, vol.55, pp. 1675-1677, Oct. 1967.

(16) Brigham, E.O., and Morrow, R.E., "The Fast Fourier Transform", IEEE Spectrum, vol. 4, No. 12, pp. 63-70, Dec. 1967.

(17) Andrews, H.C., "A high speed algorithm for the computer generation of Fourier Transforms", IEEE Trans. on Computers, vol. C-17, No.4, pp. 373, April 1968.

(18) Andrews, H.C., "Fourier Coding of Images", University of Southern California, Electronic Sciences Laboratory, USCEE Report No. 271, June 1968.

(19) Andrews, H.C., and Pratt, W.K., "Fourier Transform Coding of Images", Hawaii International Conference on System Sciences, pp. 677-679, Jan. 1968.

(20) Andrews, H.C., and Pratt, W.K., "Television Bandwidth reduction by Fourier image coding", Society of Motion Picture and Television Engineers, 103rd Technical Conference, May 1968.

(21) Andrews, H.C., and Pratt, W.K., "Television bandwidth reduction by encoding spatial frequencies", J SMPTE, vol. 77, pp. 1279-1281, Dec. 1968.

(22) Ahmed, N., Rao, K.R., and Abdussattar, A.L., "BIFORE or Hadamard Transform", IEEE Trans. Audio and Electroacoustics AU-19,No. 3, pp. 225-234, 1971.

(23) Ahmed, N., and Rao, K.R., "A Phase Spectrum for Binary Fourier Representation", Intern. J. Computer Math., Section B, vol. 3, pp. 85-101, 1971.

(24) Ohnsorg, F.R., "Binary Fourier Representation", presented at Spectrum Analysis Teccniques Sympoium, Honeywell Research Center, Hopkins,Minnesota, Sept. 20-21, 1966.

(25) Welchel, J.E. et al., "Tha Fast Fourier-Hadamard Transform and Its Use in Signal Representation and Classification", Aerospace Electronic Conference, EASCON Record, pp. 561-573, Sept. 9-11, 1968.

(26) Ahmed, N., and Cheng, S.M. "On Matrix Partitioning and a class of Algorithms", IEEE Trans. on Education, vol. E-13, pp. 103-105, Aug. 1970.

(27) Nagy, B.S., "Introduction to Real Functions and Orthogonal Expansions", New York, Oxford University Press, 1965.

(28) Pratt, W.K., Chen, W., and Welch, L.R., "Slant Transform Image Coding", IEEE Trans. on Commun., vol. COM-22, pp. 1075-1093, Aug. 1974.

(29) Ahmed, N., Natarjan, T., and Rao, K.R., "Discrete Cosine Transform", IEEE Trans. on Computers, vol. C-23, No. 1, pp. 90-93, Jan. 1974.

(30) Chen, W., Smith, C.H., and Fralick, S., "A Fast Computational Algorithm for the Discrete Cosine Transform", IEEE Trans. Commun., vol. COM-25, No. 9, pp. 1004-1009, Sept. 1977.

(31) Chen, W.H., and Smith, C.H., "Adaptive Coding of Monochrome and Color Images", IEEE Trans. Commun., vol. COM-25, No. 11, pp. 1285-1292, Nov. 1977.

(32) Paley, R.E.A.C., "On orthogonal matrices", J. Math.Phys., vol. 12, pp. 311-320, 1933.

(33) Pratt, W.K., Kane, J., and Andrews, H.C., "Hadamard Transform Image Coding", Proc. IEEE, vol. 57, pp. 58-68, Jan. 1969.

(34) Mar, H.Y.L., and Sheng, C.L., "Fast Hadamard Transform Using the H-Diagram", IEEE Trans. Computers, vol. C-22, pp. 957-959, Oct. 1973.

(35) Abdussatar, A.L., "Spectral Modes of the Walsh-Hadamard Transform", Ph.D. dissertation, Kansas State Univ., Manhattan, Kansas,USA,1972.

(36) Harmuth, H.F., "A generalized Concept of Frequency and some Applications", IEEE Trans. Inform. Theory, vol. IT-14, pp. 375-382, May 1968.

(37) Pratt, W.K., and Andrews, H.C., "Application of Fourier-Hadamard Transformation to Bandwidth Compression", in 'Picture Bandwidth Compression', Huang, T.S.,& Tretiak, O.J. Editors, New York: Gordon & Breach, pp. 515-554, 1972.

(38) Walsh, J.L., "A closed set of orthogonal functions", Am.J. Math., vol. 55, pp. 5-24, 1923.

(39) Fine, N.J., "On the Wals functions", Trans. Am. Math. Soc., vol. 65, pp. 372-414, 1949.

(40) Fine, N.J., "The generalized Walsh functions", Trans. Am. Math. Soc., vol. 69, pp. 66-77, 1950.

(41) Morgenthaler, G.W., "On Walsh-Fourier series", Trans. Am. Math. Soc., vol. 84, pp. 472-507, 1957.

(42) Henderson, K.W., "Some notes on the Walsh functions", IEEE Trans. Electronic Computers (Correspondence), vol. EC-13, pp. 50-52, Feb. 1964.

(43) Shanks, J.L., "Computation of the Fast-Walsh-Fourier Transform", IEEE Trans. Comput.(Short Notes), vol.EC-18, pp. 457-459, May 1969.

(44) Bhagavan, B.K., and Polge, R.J., "Sequencing the Hadamard Transform", IEEE Trans. Audio and Electroacoustics, vol. AU-21, pp. 472-473, Oct. 1973.

(45) Larsen, H., "An Algorithm to Compute the Sequency Ordered Walsh Transform", IEEE Trans. Acoust. Speech & Signal Process., vol. ASSP-24, No. 4, pp. 335-336, Aug. 1976.

(46) Yuen, C.K., "Remarks on the ordering of Walsh functions", IEEE Trans. Comput., vol. C-21, pp. 1452, Dec. 1972.

(47) Kunt, M., "On computation of the Hadamard Transform and the R Transform in ordered form", IEEE Trans. Comput., vol. C-24, pp. 1120-1121, Nov. 1975.

(48) Manz, J.W., "A Sequency-Ordered Fast Walsh Transform", IEEE Trans. Audio and Electroacoustics, vol. AU-20, pp. 204-205, Aug.1973.

(49) Fernandez, L.C., and Rao, K.R., "Design of a Synchronous Walsh-Function Generator", IEEE Trans. Elecromag. Comp., vol. EMC-19, pp. 407-410, Nov. 1977.

(50) Woods, J.W., and Huang, T.S., "Picture bandwidth compression by linear transformation and block quantization", in: 'Picture Bandwidth Compression', Huang, T.S., & Tretiak, O.J., Editors, New York; Gordon and Breach, pp. 555-573, 1972.

(51) Wintz, A., "Transform Picture Coding", Proc. IEEE, vol. 60,pp. 809-820, July 1972.

(52) Habibi, A., and Wintz, P.A., "Image coding by linear transformation and block quantization", IEEE Trans. Commun. Tech. , vol.COM-19, pp. 50-62, Feb. 1971.

(53) Landau, H.J., and Slepian, D., "Some computer experiments in picture processing for bandwidth reduction", Bell Syst. Tech. J., vol. 50, pp. 1525-1540, May-June 1971.

(54) Knauer, S.C., "Real-Time Video Compression Algorithm for Hadamard Transform processing", IEEE Trans. Electromag. Compat., vol. EMC-18, pp. 28-36, Feb. 1976.

(55) Walker, R., "Hadamard Transformation: A Real-Time Transformer for Broadcast Standard P.C.M. Television", BBC Research Report 1974/7, 1974.

(56) Clarke, C.K.P., "Hadamard Transformation: Walsh Spectral Analysis of Television Signals", BBC Research Report No. 1975/26, 1975.

(57) Clarke, C.K.P., "Hadamard Transformation: Assessment of bit-rate reduction methods", BBC Research Report No. RD 1976/28, 1976.

(58) Andrews, H.C., "Entropy Considerations in the Frequency Domain", Proc. IEEE Letters, vol. 56, No. 1, pp. 113-114, Jan. 1968.

(59) Dillard, G.M., "Application of Ranking Techniques to Data Compression for Image Transmission", NTC 75 Conference Record, vol. 1, pp. 22-18 to 22-22.

(60) Max, J., "Quantization for minimum distortion", IRE Trans. Inform. Theory, vol. IT-6,pp. 7-12, March 1960.

(61) Tasto, M., and Wintz, P.A., "Image coding by adaptive block quantization", IEEE Trans. Commun. Tech., vol. COM-19, pp. 957-972, Dec. 1971.

(62) Gimlett, J., "Use of Activity Classes in Adaptive Transform Image Coding", IEEE Trans. Commun., vol.COM-23, pp. 785-786, July 1975.

(63) Thorpe, P.L.M., "Contour Reduction for Digital Monochrome Image Signals", Ph.D. Thesis, University of Bradford, 1979.

(64) Kitajima, H., "Energy Packing Efficiency of the Hadamard Transform", IEEE Trans. Commun., vol. COM-24, pp. 1256-1258, Nov. 1976.

(65) Pratt, W.K., "Spatial transform coding of color images", IEEE Trans. Commun. Tech., vol. COM-19, pp. 980-992, Dec. 1971.

(66) Devereux, V.G., "Digital Video: differential coding of PAL signals based on differences between samples one subcarrier period apart", BBC Research Report BBC RD 1973/7, June 1973.

(67) Specification of television standards for 625-line system I transmissions. January 1971, BBC and ITA.

(68) Chambers, J.P., "The Use of Digital Techniques in Television Waveform Generation", IBC 1974, IEE Conf. Publ. 119, pp. 40-46, Sept. 1974.

(69) Enomoto, H., and Shibata, K., "Orthogonal transform coding system for television signals", IEEE Trans. Electromag. Compat., vol. EMC-13, pp. 11-17, Aug. 1971.

(70) Enomoto, H., and Shibata, K., et al., "Experiments on Television PCM by orthogonal transformation", 1969 Joint Convention of the I.E.C.E. of Japan, 2219.

(71) Enomoto, H., and Shibata, K., "Television signal coding method by orthogonal transformation", 6th Research Group on Television Signal Transmission. The Institute of Television Engineers of Japan, May 1968.

(72) Woods, J.W., "Video Bandwidth Compression by Linear Transformation", MIT QPR No. 91, pp. 219-224, Oct. 1968.

(73) Natarajan, T.R., and Ahmed, N., "On Interframe Transform Coding", IEEE Trans. Commun., vol. COM-25, pp. 1323-1328, Nov. 1977.

(74) Knab, J.J., "Effects of Round-Off Noise on Hadamard Transformed Imagery", IEEE Trans. Commun., vol. COM-25, pp. 1292-1294, Nov. 1977.

(75) Kassam, S.A., "Quantization Based on the Mean-Absolute-Error Criterion", IEEE Trans. Commun., vol. COM-26, pp. 267-270, Feb. 1978.

(76) Netravali, A.N., Prasada, B., and Mounts, F.W., "Some Experiments in Adaptive and Predictive Hadamard Transform Coding of Pictures", Bell Syst. Tech. J., vol. 56, No.8, pp. 1531-1547, Oct. 1977.

(77) Mounts, F.W., Netravali, A.N., and Prasada, B., "Design of Quantizers for Real-Time Hadamard-Transform Coding of Pictures", Bell Syst. Tech. J., vol. 56, No.1, pp. 21-48, Jan. 1977.

(78) Prasada, B., Mounts, F.W., and Netravali, A.N., "Level Reassignment: 'A Technique for Bit Rate Reduction' ", Bell Syst. Tech.J., vol. 57, No.1, pp. 61-73, Jan. 1978.

(79) Roese, J.A., "Interframe Coding of Digital Image Using Transform and Hybrid Transform/Predictive Techniques", Tech.Report No. USCIPI 700, University of Southern California, June 1976.

(80) Ishii, M., "Picture Bandwidth Compression by DPCM in the Hadamard Transform Domain", Fujitsu Sci.&Tech. J., 10, No. 3, pp. 51-65, Sept. 1974.

(81) Limb, J.O., "Picture Coding: The use of viewer model in source encoding", Bell Syst. Tech. J., vol. 52, No. 8, pp. 1271-1302, Oct. 1973.

(82) Ishii, M., Sasaki, S., and Kojima, T., "Picture Bandwidth Compression by DPCM in the Hadamard Transform Domain", Proc. 7th Hawaii International Conf. on System Sciences, pp. 10-12, Jan. 1974.

(83) Tasto, M., and Wintz, "Picture bandwidth compression by adaptive block quantization", School of Electrical Engineering, Purdue University, Lafayette, Ind., Tech.Rep. TR-EE 70-14, June 1970.

(84) Tsto, M., Habibi, A., and Wintz, P.A., "Adaptive and non-adaptive image coding by linear transformation and block quantization", presented at the Symp. Picture Coding, North Carolina State University, Raleigh, N.C., Sept. 10, 1970.

(85) Claire, E.J., "Bandwidth reduction in image transmission", in Conf. Rec., 1972 IEEE Int. Conf. Communications, pp. 39-8 to 39-13.

APPENDICES

A.1. Program segment for computing Directing Index D1.

```
      ITOT=0
      INDEX=0
      DO3 K=1,NSG
      ISUM(K)=0
      DO4I1=IS(K),IE(K)
      ISUM(K)=ISUM(K)+IABS(ID(I1))
4     CONTINUE
      IF(ISUM(K)-ITH(K)) 91,91,92
91     INDX(K)=LOW(K)
      GOTO93
92    INDX(K)=IHGH(K)
93    ITOT=ITOT+ISUM(K)
      INDEX=INDEX+INDX(K)
3     CONTINUE
      J=INDEX
```

J = the directing index value after comprtation.

Notes :

ID values of coefficients (array of 16).

NSG total number of subvectors.

IS order of the first coefficient in a subvector.

IE order of the last coefficient in the subvector.

ITH the threshold value corresponding to a subvector,(or H2 if its activity is to be considered).

LOW the weighting factor corresponding to low activity.

IHGH the weighting factor corresponding to high activity.

This segment is for the general computation of D1, whether H2 is included or not. This is achieved by assigning the appropriate values for NSG, IS, and IE in each case.

A.2. Program segment for computing Directing Index D2.

```
      INDEX=0
      INDX(2)=0
      IF(IABS(ID(2)).GT.IT2) INDX(2)=6
      SMODD=FLOAT(IABS(ID(5))+IABS(ID(7)))
      SMEV=FLOAT(IABS(ID(6))+IABS(ID(8)))
      TOTM=SMODD+SMEV
      IF(TOTM.EQ.0.0) GOTO2001
      SMODD=0.0
       DO3001IOD=9,15,2
3001  SMODD=SMODD+FLOAT(IABS(ID(IOD)))
      SMEV=0.0
      DO3002IEV=10,16,2
3002  SMEV=SMEV+FLOAT(IABS(ID(IEV)))
      RMED=AMAX1(SMODD,SMEV)
      RHIGH=AMAX1(SMODD,SMEV)
      BALANC=RHIGH/RMED
      IF(BALANC.LE.THIN) GOTO 2002
      IF(BALANC.GE.THAX) GOTO 2001
      INDX(3)=2
      IF(RMED.EQ.SMEV) INDX(3)=4
      INDX(4)=1
      IF(RHIGH.EQ.SHEV) INDX(4)=2
      INDEX=INDX(3)+INDX(4)
      GOTO2005
2001  INDEX=2
      GOTO 2005
2002  INDEX=1
2005  J=INDEX+INDX(2)
```

Notes (in addition to notes on segment for D1) :

IT2 threshold value of H2 (in case of Double Range Coder).

SMODD sum of absolute values in odd subset of medium subvector.

SMEV sum of absolute values in even subset of medium subvector.

SHODD sum of absolute values in odd subset of high subvector.

SHEV sum of absolute values in even subset of high subvector.

RMED activity measure of the whole medium subvector,(representative).

RHIGH activity measure of the whole high subvector, (representative).

BALANC balance between medium and high subvector activities.

This segment is, like for D1, for general computation of D2, including H2.

A.3. Program segment for computing Directing Index D3.

```
      INDEX=0
      INDX(2)=0
      IF(IABS(ID(2)).GT.IT2)  INDX(2)=6
      SMODD=FLOAT(IABS(ID(5))+IABS(ID(7)))
      SMEV=FLOAT(IABS(ID(6))+IABS(ID(8)))
      TOTM=SMODD+SMEV
      IF(TOTM.EQ.0.0)  GOTO2001
      SHODD=0.0
       DO3001IOD=9,15,2
3001  SHODD=SHODD+FLOAT(IABS(ID(IOD)))
      SHEV=0.0
      DO3002IEV=10,16,2
3002  SHEV=SHEV+FLOAT(IABS(ID(IEV)))
      SHUP=0.0
      DO 111 I=13,16
111   SHUP=SHUP+FLOAT(IABS(ID(I)))
      SHLO=0.0
      DO 222 I=9,12
222   SHLO=SHLO+FLOAT(IABS(ID(I)))
      RMED=AMAX1(SMODD,SMEV)
      RHIGH=AMAX1(SHODD,SHEV,SHUP,SHLO)
      BALANC=RHIGH/TOTM
      IF(BALANC.LE.TMIN) GOTO 2002
      IF(BALANC.GE.TMAX) GOTO 2001
      INDEX=3
      IF(RHIGH.EQ.SHLO) GOTO 2005
      IF(RHIGH.EQ.SHUP) GOTO 2007
      IF(RHIGH.EQ.SHODD) GOTO 2009
      IF(RHIGH.EQ.SHEV) GOTO2011
2007  INDEX=4
      GOTO 2005
2009   INDEX=5
      GOTO 2005
2011   INDEX=6
      GOTO2005
2001  INDEX=2
      GOTO 2005
2002  INDEX=1
2005  J=INDEX+INDX(2)
```

Notes (in addition to notes on D1 and D2) :

SHLO sum of absolute values in lower subset of high sequency subvector.

SHUP sum of absolute values in upper subset of the same subvector.

Again, it is to be noted that this segment computes D3, including the two ranges of H2.

A.4. Program segment for computing Directing Index C1.

```
      I10=IABS(ID(10))+IABS(ID(11))+IABS(ID(12))
      ITOT=0
      INDEX=0
      DO3 K=1,NSG
      ISUM(K)=0
      DO4I1=IS(K),IE(K)
      ISUM(K)=ISUM(K)+IABS(ID(I1))
4     CONTINUE
      IF(K.EQ.3) ISUM(K)=ISUM(K)-IABS(ID(6))
      IF(K.EQ.4) ISUM(K)=ISUM(K)-I10
      IF(ISUM(K)-ITH(K)) 91,91,92
91     INDX(K)=LOW(K)
      GOTO93
92    INDX(K)=IHGH(K)
93    ITOT=ITOT+ISUM(K)
      INDEX=INDEX+INDX(K)
3     CONTINUE
      J=INDEX
```

A.5. Program segment for computing Directing Index C2.

```
      ITOT=0
      INDEX=0
      DO3 K=1,NSG
      ISUM(K)=0
      DO4I1=IS(K),IE(K)
      ISUM(K)=ISUM(K)+IABS(ID(I1))
4     CONTINUE
      IF(K.EQ.3) ISUM(K)=ISUM(K)-IABS(ID(8))
      IF(K.EQ.4) ISUM(K)=ISUM(K)-IABS(ID(9))
      IF(ISUM(K)-ITH(K)) 91,91,92
91     INDX(K)=LOW(K)
      GOTO93
92    INDX(K)=IHGH(K)
93    ITOT=ITOT+ISUM(K)
      INDEX=INDEX+INDX(K)
3     CONTINUE
      J=INDEX
```

Notes : see notes on segment for D1.

A.6. FORTRAN IV Listing of a Sample Program Simulating Directing Index D1, Transforming and Coding.

```
0          LIBRARY (SUBGROUPBRAD)
1          LIBRARY (OWLIB)
2          PROGRAM (QUANTIZING)
3          INPUT1=TR0
4          INPUT3=TR3
5          INPUT4=TR4
6          OUTPUT2=LP0
7          COMPRESS INTEGER AND LOGICAL
8          END
9          MASTER QUANT
10         DIMENSION ID(16),IDS(16),IX(16),IY(16),IPR(16),
11        *IDD(9,16),IR(8,16),IDDC(9,16,16),IRR(8,16,16),
12        *ERQ(16),INR(16),ERTN(16),
13        *ERT(16),ERQN(16),
14        *PST(16),IS(4),IE(4),LOW(4),ITH(4),[illegible],ITH(4),
15        *IHGH(4),INDX(4),ISUI(4),NVS(16),XERT(16),XERTN(16),
16        *XERQ(16),XERQN(16)
17         NSG=4
18         NUM=16
19         JTQ=16
20         IS(1)=2
21         NL=6
22         NM=9
23 C
24 C
25 C          NSG   = NUMBER OF SUBVECTORS
26 C          NUM   = VECTOR SIZE
27 C          JTQ   = MAXIMUM NUMBER OF QUANTIZING SCHEMES
28 C          IS    = ORDER OF STARTING COEFFICIENT OF A SUBVECTOR
29 C          IE    = ORDER OF LAST COEFFICIENT IN A SUBVECTOR
30 C          NL    = NUMBER OF QUANTIZING LEVELS (DECISION LEVELS)
31 C          NM    = NUMBER OF REPRESENTATIVE VALUES
32 C
33 C
34 C
35         DO 1 K=1,NSG
36 1       IE(K)=2**K
37         DO 2 K=2,NSG
38 2       IS(K)=IE(K-1)+1
39         RMIN=0.1E-9
40         READ(1,120)SN, NVT,ITH,LOW,IHGH
41 C
42 C
43 C          SN        = LOWEST SNR (ONLY FOR COMPUTING EXC.SNR)
44 C          NVT       = TOTAL NUMBER OF VECTORS
45 C          ITH       = ARRAY OF THRESHOLDS FOR DIFFERENT SUBVECTORS
46 C          LOW       = ARRAY OF WEIGHTING FACTORS FOR LOW ACTIVITY
47 C          IHGH      = ARRAY OF WEIGHTING FACTORS FOR HIGH ACTIVITY
48 C
49 C
```

```
      WRITE(2,601)NVT,SN,ITH
      DO 222 J=1,JTQ
      DO 222 J=1,JTQ
      DO 555 I=2,NUM
      READ(1,250) IDDC(K,I,J)
      READ(1,350) IRR(K,I,J)
  222 CONTINUE
  555 CONTINUE
  222 CONTINUE
C
C
C     READING THE CODING PARAMETERS :
C     IDDC     = DECICION (QUANTIZING) LEVELS
C     IRR      = REPRESENTATIVES
C
C
      SERDN=0.0
      SERQN=0.0
      SERTN=0.0
      SERD=0.0
      SERT=0.0
      SERQ=0.0
      TSN=0.0
      SERTX=0.0
      SERQX=0.0
      SERTNX=0.
      SERQNX=0.0
      PTQ=0.0
      PTT=0.0
      PTR=0.0
C
C
C
C     ERD    ERRORS DUE TO NORMALIZATION ONLY
C     ERQ    ERRORS DUE TO QUANTIZATION ONLY
C     ERT    TOTAL ERRORS DUE TO BOTH
C
C
C
C      S    PRECEDING ANY EXPRESSION MEANS THE SUM OF ERRORS
C           ALLOVER THE WHOLE IMAGE.
C      N    FOLLOWING ANY EXPRESSION MEANS THE P-P CRITERION
C      X    IN ANY ERROR EXPRESSION MEANS EXC. CRITERION
C      ( )  MEANS A CRITERION FOR A PARTICULAR INDEX VALUE
C
C
      DO 20 I=1,JTQ
      ERT(I)=0.0
      ERQ(I)=0.0
      ERTN(I)=0.0
      ERQN(I)=0.0
      XERT(I)=0.0
      XERQ(I)=0.0
      XERTN(I)=0.0
      XERQN(I)=0.0
   20 CONTINUE
```

```
      DO 2000 IND=1,NVT
C
C        ID       = ARRAY OF SAMPLES
C
      READ(3,100) ID
      DO 111 I=1,NUM
      IDS(I)=ID(I)
111   CONTINUE
C
C        IDS      - AN ARRAY OF SAVED VALUES OF SAMPLES FOR
C                   LATER COMPARISONS AND ERROR COMPUTATION
C
C
      CALL HADAM (NUM,ID,IY)
C
C        HADAMARD TRANSFORMING OF ORIGINAL DATA
C        ID  ON EXIT WILL HAVE THE TRANSFORM COEFFICIENTS
C
      CALL ROUND (ID,NUM,NUM)
C
C        COEFFICIENTS ARE PRE-NORMALIZED BY A FACTOR EQUALS TO
C        THE WHOLE VECTOR SIZE 'NUM'.
C        ON EXIT ID WILL BE AN ARRAY OF NORMALIZED AND ROUNDED
C        OFF COEFFICIENTS
C
C
C          ****   COMPUTING DIRECTING INDEX DI   ****
C
C
      ITOT=0
      INDEX=0
      DO3 K=1,NSG
      ISUM(K)=0
      DO4I1=IS(K),IE(K)
      ISUM(K)=ISUM(K)+IABS(ID(I1))
4     CONTINUE
      IF(ISUM(K)-ITH(K)) 91,91,92
91    INDX(K)=LOW(K)
      GOTO 93
92    INDX(K)=IHGH(K)
93    ITOT=ITOT+ISUM(K)
      INDEX=INDEX+INDX(K)
3     CONTINUE
      J=INDEX
C       ****      J=  THE DIRECTING INDEX     ****
C
      NVS(J)=NVS(J)+1
C
C        NVS     = NUMBER OF VECTORS IN A PARTICULAR CLASS INDEX
C
      DO 1700 I=1,NUM
1700  IX(I)=ID(I)
C
C        SAVING THE NORMALIZED COEFFICIENTS
```

```
C
      CALL HADAM(NUM,IX,IY)
C
C        INVERSE TRANSFORMING OF NORMALIZED COEFFICIENTS
C        ON EXIT, IX  WILL BE AN ARRAY OF RECONSTRUCTED
C        SAMPLES AFTER SUBJECTING THE COEFFICIENTS TO
C        NORMALIZING ONLY
C
C
      RDV=0.0
      CALL LNMSE(NUM,IDS,IX,RDV)
C
C        COMPUTING LMSE DUE TO NORMALIZATION ONLY
C
      SERD=SERD+RDV
      IF(RDV.EQ.0.0) RDV=RMIN
C
C           NO NORMALIZATION ERROR AT ALL
C
      SNRD=-10*ALOG10(RDV)
      IF(SNRD.GE.99.0) PTR=PTR+1.0
      CALL ERROR(NUM,IDS,IX,P)
C
C           COMPUTING ERROR DUE TO NORMALIZATION
C
      SERDN=SERDN+P
      DO 12 I=1,NUM
      INR(I)=IX(I)
      IX(I)=ID(I)
12    CONTINUE
C
C
C        SELECTING THE CODING PARAMETERS OF THE APPROPRIATE
C        CODER ASSOCIATED WITH THE INDEX VALUE 'J'
C
C
      DO 444 K2=2,NUM
      DO 333 K1=1,NL
      IR(K1,K2)=IRR(K1,K2,J)
      IDD(K1,K2)=IDDC(K1,K2,J)
333   CONTINUE
444   CONTINUE
      IDD(NM,K2)=IDDC(NM,K2,J)
C
C
C        CODING THE NORMALIZED COEFFICIENTS
C
C
      CALL ICODER (2,NUM,NL,NM,IX,IDD,IR)
C
C
C        IX  ON EXIT HAVE THE CODED COEFFICIENTS
C
C
C
C        INVERSE TRANSFORMING AFTER CODING
C
C
C
      CALL HADAM(NUM,IX,IY)
```

```
C
C          IX    AN ARRAY OF THE RECONSTRUCTED DATA AFTER
C                 NORMALIZING AND CODING
C
C
C             COMPUTING THE ERROR CRITERIA FOR EACH VECTOR
C
C
      SER=0.0
      CALL LNMSE(NUM,IDS,IX,SER)
      ERT(J)=ERT(J)+SER
      IF(SER.EQ.0.0) SER=RMIN
      SNRT=-10*ALOG10(SER)
      CALL ERROR(NUM,IDS,IX,P)
      ERTN(J)=ERTN(J)+P
      SER2=0.0
      CALL LNMSE(NUM,INR,IX,SER2)
      ERQ(J)=ERQ(J)+SER2
      IF(SER2.EQ.0.0) SER2=RMIN
      SNR=-10*ALOG10(SER2)
      CALL ERROR(NUM,INR,IX,P2)
      ERQN(J)=ERQN(J)+P2
      IF(SNRT.GT.SN) GOTO 1999
      WRITE(2,805) IND,J,SNRT,SNR
      IEXP=IEXP+1
C
C       IEXP     = NUMBER OF EXCLUDED CASES FOR EXC.SNR
C
      XERT(J)=XERT(J)+SER
      XERTN(J)=XERTN(J)+P
      XERQ(J)=XERQ(J)+SER2
      XERQN(J)=XERQN(J)+P2
 1999 CONTINUE
C
C       COUNTING FOR NO QUANTIZING ERROR
C       COUNTING FOR NO TOTAL ERROR
C
      IF(SNR.GE.99.0) PTQ=PTQ+1.0
      IF(SNRT.GE.99.0) PTT=PTT+1.0
 2000 CONTINUE
      DO 2001 J=1,JTQ
C
C
C          COMPUTING THE OVERALL ERROR CRITERIA FOR EACH INDEX
C
C          S   IS FOR THE SUI
C          Q   IS FOR THE QUANTIZING (CODING)
C          T   IS FOR TOTAL, I.E. NORMALIZATION+CODING
C          X   IS FOR THE EXCLUSIVE CRITERIA
C        N   IS FOR THE PPSNR CRITERION
C
C
      SERQN=SERQN+ERQN(J)
      SERTN=SERTN+ERTN(J)
      SERQ=SERQ+ERQ(J)
      SERT=SERT+ERT(J)
      SERQNX=SERQNX+XERQN(J)
      SERTNX=SERTNX+XERTN(J)
      SERQX=SERQX+XERQ(J)
      SERTX=SERTX+XERT(J)
 2001 CONTINUE
C
```

```
279 C
280 C        COMPUTING THE OVERALL ERROR CRITERIA FOR THE WHOLE IMAGE
281 C
282         NX=NVT-IEXP
283         SERTX=SERT-SERTX
284         SERQX=SERQ-SERQX
285         SERTX=-10*ALOG10(SERTX/NX)
286         SERQX=-10*ALOG10(SERQX/NX)
287         DEN1=255.*255.*1.X
288         SERQNX=(SERQN-SERQNX)/DEN1
289         SERTNX=(SERTN-SERTNX)/DEN1
290         SERD=-10*ALOG10(SERD/NVT)
291         SERT=-10*ALOG10(SERT/NVT)
292         SERQ=-10*ALOG10(SERQ/NVT)
293         DEN=255.*255.*NVT
294 C
295 C           MAX. VALUE OF LUMINANCE = 255 LEVELS.
296 C
297         SERDN=SERDN/DEN
298         SERQN=SERQN/DEN
299         SERTN=SERTN/DEN
300         WRITE(2,300)
301         WRITE(2,400) SERD,SERT,SERQ,SERDN,SERTN,SERQN
302         WRITE(2,806) IEXP,SERTX,SERQX,SERTNX,SERQNX
303         WRITE(2,500)
304         DO18J=1,JTQ
305         PST(J)=100.0*FLOAT(NVS(J))/NVT
306         IF(ERT(J).EQ.0.0) ERT(J)=RMIN
307         IF(NVS(J).EQ.0) GOTO77
308         ERT(J)=-10.*ALOG10(ERT(J)/NVS(J))
309         IF(ERQ(J).EQ.0.0) ERQ(J)=RMIN
310         ERQ(J)=-10.0*ALOG10(ERQ(J)/NVS(J))
311         GOTO75
312 77      ERT(J)=0.0
313         ERQ(J)=0.0
314 75      CONTINUE
315         IF(NVS(J).EQ.0) GOTO 79
316         DEN=255.*255.*NVS(J)
317         ERTN(J)=ERTN(J)/DEN
318         ERQN(J)=ERQN(J)/DEN
319 79      CONTINUE
320         WRITE(2,600) J,PST(J),ERT(J),ERQ(J),ERTN(J),ERQN(J)
321 18      CONTINUE
322         PTR=100.0*PTR/NVT
323         PTQ=100.0*PTQ/NVT
324         PTT=100.0*PTT/NVT
325 C
326 C
327 C           PTR   PERCENT OF NO NORMALIZATION ERROR AT ALL
328 C           PTQ   PERCENT OF NO QUANTIZATION ERROR AT ALL
329 C           PTT   PERCENT OF NO ERRORS AT ALL
330 C
331 C
332         WRITE(2,801) PTR
333         WRITE(2,802) PTQ
334         WRITE(2,803) PTT
```

```
335 250    FORMAT(9I0)
336 350    FORMAT(8I0)
337 801    FORMAT(///,10X,'NO ROUND OFF NOISE = ',F6.2,3X,'PERCENT')
338 802    FORMAT(/,10X,'NO QUANTIZING NOISE= ',F6.2,3X,'PERCENT')
339 803    FORMAT(/,10X,'NO NOISE AT ALL     = ',F6.2,3X,'PERCENT')
340 804    FORMAT(///,10X,'VECTOR',8X,'INDEX',5X,'TOTAL NOISE DB',
341       *6X,'QUANT NOISE DB',//)
342 805    FORMAT(10X,I5,10X,I3,9X,F8.3,12X,F8.3)
343 806    FORMAT(///,8X,'EXCEPT',I6,'VECTORS',4X,F6.2,9X,F6.2,21X,
344       *F10.8,5X,F10.8,//)
345 100    FORMAT(1610)
346 300    FORMAT(///,13X,'ROUND DB',7X,'TOTAL DB',7X,'QUANT DB',
347       *5X,'ROUND NORM',5X,'TOTAL NORM',5X,'QUANT NORM',/)
348 400    FORMAT(7X,3(9X,F6.2),1X,3(5X,F10.8))
349 500    FORMAT(///,5X,'INDEX',5X,'PROBABILITY',8X,'TOTAL DB',8X,
350       *'QUANT DB',6X,'TOTAL NORM',6X,'QUANT NORM',/)
351 600    FORMAT(I10,3(8X,F8.3),2(6X,F10.8))
352 601    FORMAT(//,I6,'VECTORS',5X,'SN=',F6.1,'  DB',
353       *10X,'THRESHOLDS',5X,4I6,//)
354 702    FORMAT(1X,'NUMB.',5X,'OFF DB',,' IN DB',6X,16I2,4X
355       *,6X,16I2,4X,'IN DB',//)
356 120    FORMAT(F0.0,1310)
357        STOP
358        END
359        FINISH
```

A.7. Subroutines Used in the Simulation Program of A.6.

```
 0        SUBROUTINE ERROR(N,IX,IY,P)
 1 C
 2 C
 3 C         THIS SUBROUTINE CALCULATES THE MEAN-SQUARED-ERROR
 4 C         OVER A VECTOR ARRAY.
 5 C         N      = THE VECTOR SIZE
 6 C         IX     = ARRAY OF ORIGINAL VALUES
 7 C         IY     = ARRAY OF RECONSTRUCTED VALUES
 8 C         P      = ON EXIT I WILL HAVE THE MEAN-SQUARED-ERROR
 9 C
10 C
11        DIMENSION IX(N),IY(N)
12        P=0.0
13        DO1 L=1,N
14        ER=IX(L)-IY(L)
15        ER=ER*ER
16        P=P+ER
17 1      CONTINUE
18        P=P/N
19        RETURN
20        END
```

Note. For subroutine HADAM, see for example, Reference 10.

```
      SUBROUTINE ROUI D(ID,NUM,I!)
C
C
C       THIS SUBROUTINE I ORMALIZES THE COEFFICIENTS IN
C       THE TRANSFORI DOI AIN TO THE RANGE OF ORIGINAL
C       SAMPLE VALUES
C       IT COULD BE USED ALSO FOR ANY TYPE OF NORMALIZATION
C       I.E. PRE-, POST-, OR DOUBLE HALF NORMALIZATION.
C
C       ID     = ARRAY OF COEFFICIENTS
C       NUM    = VECTOR SIZE
C          I   = NORMALIZING FACTOR. THIS CAN BE ADJUSTED TO
C                SUIT ANY PARTICULAR TYPE OF NORMALIZATION. IN
C                THE CASES OF PRE-,AND POST-, M=NUM.
C
C
      DIMENSION ID(NUI')
      DO1 I=1,NUM
      IFACT=1
      IF(ID(I).LT.0) IFACT=-1
      ID(I)=IFACT*(IABS(ID(I))+M/2)/M
1     CONTINUE
      RETURN
      END


      SUBROUTINE LNMSE(N,IX,IY,PER)
C
C       THIS SUBROUTINE COMPUTES THE LOCALLY NORMALIZED
C       LEAN-SQUARED-ERROR.
C
C      DIMENSION IX(N),IY(N)
C
C       N      = VECTOR SIZE
C       IX     = ORIGINAL SAMPLES ARRAY
C       IY     = RECONSTRUCTED ARRAY VALUES
C       PER    = ON EXIT WILL HAVE THE LOCALLY NORMALIZED
C                MEAN-SQUARE -ERROR
C
C
      PER=0.0
      DO 1 L=1,N
      ER=IX(L)-IY(L)
      IF(IX(L).EQ.0)  GOTO1
      ER=ER/IX(L)
      ER=ER*ER
      PER=PER+ER
1     CONTINUE
      PER=PER/N
      RETURN
      END
```

```
      SUBROUTINE ICODER(IS,IEND,NUM,NL,NM,IX,IDD,IR)
      DIMENSION IX(NUM),IDD(NM,NUM),IR(NL,NUM)
C
C
C       THIS SUBROUTINE IS FOR CODING INTEGER COEFFICIENTS
C        PARAMETERS ARE AS FOLLOWS
C
C        IS     =ORDER OF STARTING COEFFICIENT
C        IEND   = ORDER OF LAST COEFFICIENT
C        NUM= MAX. NUMBER OF TOTAL COEFF.
C        NL =NUMBER OF REPRESENTATIVE LEVELS
C        NM =NUMBER OF DECISION LEVELS(=NL+1)
C        IX =COEFF. ARRAY (NUM)
C           ON ENTRY, THIS WILL HAVE THE ACTUAL VALUES.
C           ON EXIT, THEY WILL HAVE THE CODED VALUES.
C         IDD =DECISION ARRAY (NM,NUM)
C         IR =REPRESENTATIVES ARRAY (NL,NUM)
C
C
C
      DO 100 I=IS,IEND
      IP=1
      IN=1
C
C          TRY FOR THE FIRST QUANTIZING STEP
C
      IF(IX(I).LT.0) IP=-1
      IX(I)=IABS(IX(I))
   10 IF(IX(I).GE.IDD(IN,I).AND.IX(I).LT.IDD(IN+1,I))
     * IX(I)=IR(IN,I)
      IF(IX(I).EQ.IR(IN,I)) GOTO 99
C
C          IF THE CORRECT STEP HAS BEEN FOUND, THEN NEXT COEFF.
C          IF NOT, TRY NEXT STEP.
C
      IN=IN+1
      GOTO 10
   99 IX(I)=IP*IX(I)
  100 CONTINUE
      RETURN
      END
```